RFnest™ User Manual

Intelligent Automation, Inc.

Version 2.11.6

RFview 2.11+, D5-series (rev1), and A208

www.i-a-i.com/rfnest

15400 Calhoun Drive, Suite 190, Rockville, MD 20855 | Phone: 301-294-5200 | Fax: 301-294-5201

Revision History

Version	Date	Author	Section(s)	Description
2.1	10/20/14	jyackoski	6.3	Clarified Doppler shift calculation
2.2	10/28/14	nlenzi	11.11, 6.3	Added section for command-line interface. Added per-tap Doppler Boolean to resource profile
2.2.1	11/20/14	Nlenzi/jyackoski	11.7	Updated delay profile file format based on API rev 2.2
2.2.2	1/28/15	jyackoski	11.9, 13	Added scenario script feature
2.2.3	1/29/15	jyackoski	6.3, 8.2.2, 8.2.4, 8.3.5	Updated resource profiles, updated radio type to explain offsets and orientations
2.2.4	2/19/15	nlenzi	11.6.2, 11.9.3, 11.9.5	Update scenario script commands, updated event exporter examples
2.2.5	4/1/15	nlenzi	11.9.3	Update scenario script commands
2.2.6	3/31/15	jyackoski	8.4, 9.3.4, 11.4, 11.7.2.7	Add Automatic Gain Control, cleanup formatting
2.2.7	4/11/15	jyackoski	2.1	Add supporting equipment list
2.11.6	5/4/15	jyackoski	all	Cleanup setup instructions, revise all diagrams and descriptions to 2.11.6

TABLE OF CONTENTS

1 Introduction 1

2 Prerequisites 1
- 2.1 Supporting Equipment . . . 1
- 2.2 Linux OS set up / supporting software configuration instructions . . . 2
- 2.3 Windows OS Setup . . . 2
- 2.4 Java installation instructions . . . 2
- 2.5 CORE installation (OPTIONAL) . . . 3
- 2.6 EMANE Installation: . . . 3
- 2.7 Libxml2 install . . . 3

3 RFnest Software Installation 4
- 3.1 RFview Installation . . . 4
- 3.2 CEC installation . . . 4

4 CEC Configuration 4

5 Starting the System 5
- 5.1 Basic Cabling (A-Series) . . . 5
- 5.2 Basic Cabling (D-Series) . . . 5
- 5.3 Resource Profiles (D-Series Only) . . . 6
- 5.4 Network Startup . . . 6
- 5.5 Running the CEC . . . 7
- 5.6 RFview User Interface Startup . . . 7
- 5.7 Selecting a Resource Profile (D-Series Only) . . . 8
- 5.8 Verifying System Operation . . . 8

6 Optional Configuration 8
- 6.1 Compiling RFview . . . 8
- 6.2 Impact and Driver Installation . . . 9
- 6.3 RFview Configuration . . . 9
- 6.4 RFview Map Configuration . . . 9
- 6.5 Pre-caching RFview Map Data . . . 10
- 6.6 Generating Tactical Graphics for use in RFview . . . 10
- 6.7 Additional CEC Standard Mode Parameters . . . 11
- 6.8 CEC Test mode parameters . . . 11
- 6.9 CEC Eventdaemon.xml Configuration . . . 12
- 6.10 Loading the Hardware (D-Series Only) . . . 12

7 Basic Concepts 13
- 7.1 Major System Components . . . 13
- 7.2 Typical Workflow . . . 13
- 7.3 Platform Types and Radio Types . . . 14
- 7.4 Groups and Frequency Groups . . . 14
- 7.5 Maps and Overlays . . . 15
- 7.6 Scenario Analysis, Recording, and Demonstration . . . 15

8 Creating the Basic Scenario Attributes 15
- 8.1 Platform Type Editor . . . 15
  - 8.1.1 Create platform type . . . 16
  - 8.1.2 Edit platform type . . . 17
  - 8.1.3 Delete platform type . . . 18
- 8.2 Radio Type Editor . . . 19
  - 8.2.1 Create Radio Type . . . 19
  - 8.2.2 Edit radio . . . 21

8.2.3 Delete radio . . . 22
8.2.4 Radio Coordinate System . . . 22
8.3 Platforms Editor . . . 22
8.3.1 Create new platform . . . 22
8.3.2 Delete platform . . . 23
8.3.3 Edit platform . . . 24
8.3.4 Add radio to platform . . . 25
8.3.5 Placing Multiple Radios at Different Locations on a Platform . . . 26
8.4 Group Configuration . . . 26
8.5 Map controls/options . . . 29
8.6 Overlay Tab . . . 30
8.7 Platforms Overlay . . . 31
8.8 Gain vs Distance Overlay . . . 32
8.9 Link Quality Overlay . . . 35
8.10 Waypoint Overlay . . . 37
8.11 Platform Routes Overlay . . . 37
8.12 Links/Channel Properties Editor . . . 38
8.13 Route Editor . . . 40
8.14 Saving/Loading scenarios . . . 41
8.15 Scenario Record/Replay . . . 42

9 Attaching, Verifying, Calibrating, and Operating Radios in the System 42
9.1 Port Configuration Summary . . . 43
9.2 Port Status (D-Series Only) . . . 43
9.3 Alerts . . . 44
9.3.1 Create 'Tx Stopped' alert (D-Series Only) . . . 45
9.3.2 Create 'Rx Stopped' alert (D-Series Only) . . . 46
9.3.3 Create 'DCU Notification Stopped' alert . . . 47
9.3.4 Create 'Automatic Gain Control Out of Range' alert . . . 48
9.4 Recording and Replaying Signals Using the Signal Tool (D-Series Only) . . . 49
9.4.1 Examples of Recorded Signals . . . 51
9.5 Attaching Radios . . . 52
9.6 Choosing External Attenuation, Setting Gains, and Calibrating Radios . . . 54
9.6.1 Determining if the radios will be Isolated . . . 54
9.6.2 Ensuring the input signal is at the correct level (D-Series Only) . . . 54
9.6.3 Adjusting the Transmit Power Offset . . . 55
9.6.3.1 Dynamic range method . . . 55
9.6.3.2 Scenario based method . . . 56
9.7 Verifying Radio Operation and Isolation . . . 56
9.7.1 Verifying Radio Operation . . . 56
9.7.2 Verifying Radio Isolation . . . 57
9.8 Configuring Thresholds for Scenario Layout/Display . . . 58

10 Basic Monitoring and Data Collection 59
10.1 RSSI/Link Quality . . . 59
10.2 Routing/Communication Events . . . 60
10.2.1 Packet Routing Overlay . . . 60
10.2.2 Usage in a Scenario . . . 61
10.3 Real-time Graphs . . . 62

11 Advanced Analysis Tools and Topics 63
11.1 Adding Point Source Emitters and Background Noise (D-Series Only) . . . 63
11.1.1 Point Source Emitters . . . 64
11.1.2 Background Noise . . . 64
11.2 Events Editor . . . 64
11.2.1 Create event . . . 65
11.2.2 Create Signal Tool Event (D-Series Only) . . . 65

11.2.3 Create Position/Antenna Event . . . 67
11.2.4 Create Channel Event . . . 69
11.2.5 Edit Event . . . 70
11.2.6 Delete Event . . . 70
11.3 Swimlane (D-Series Only) . . . 70
11.4 Automatic Gain Control (D-Series Only) . . . 72
11.4.1 Using AGC . . . 72
11.4.2 AGC Monitor . . . 74
11.4.3 AGC Behavior Example . . . 75
11.4.4 AGC Alerts and Logging . . . 76
11.4.5 AGC Caveats . . . 76
11.5 Antenna Pattern Files . . . 78
11.6 Virtual node configuration. . . . 79
11.7 Event Deck . . . 79
11.7.1 EventDeckExporter Usage . . . 79
11.7.2 Event Formats . . . 80
11.7.2.1 Position Update Event . . . 80
11.7.2.2 RSSI Event . . . 81
11.7.2.3 Packet Route Event . . . 81
11.7.2.4 Remove Packet Route Event . . . 82
11.7.2.5 DCU Notification Type A Event . . . 82
11.7.2.6 Perf Data Event . . . 83
11.7.2.7 Automatic Gain Control Event (D-Series Only) . . . 83
11.7.2.8 Signal Status Update Event (D-Series Only) . . . 83
11.7.2.9 Swimlane Event (D-Series Only) . . . 84
11.7.2.10 Set Radio Parameters Event . . . 85
11.7.2.11 Manual Channel Event . . . 85
11.7.2.12 GeometryChangeRequest . . . 86
11.7.2.13 SignalToolScriptEvent (D-Series Only) . . . 87
11.7.2.14 DelayProfileScriptEvent (D-Series Only) . . . 87
11.8 Delay Profile Format (D-Series Only) . . . 88
11.9 Signal Recording Format (D-Series Only) . . . 88
11.10 Import Scenario Script . . . 89
11.10.1 Create Platform . . . 90
11.10.2 Create Radio . . . 90
11.10.3 Update Platform . . . 91
11.10.4 Update Link . . . 91
11.10.5 Set Route . . . 92
11.11 Import AWSIM Scenario . . . 93
11.12 Kinematic Trajectory Generator (KTG) . . . 93
11.13 Command Line Interface . . . 94

12 Built-in test/Self-test and RF Chain Calibration (D-Series Only) 94
12.1 Using the Built-in Test Dialogs . . . 94
12.2 Running the DDB Sync Test . . . 95
12.2.1 Handling Failure . . . 96
12.3 Running the CV Doppler Test . . . 96
12.3.1 Handling Failure . . . 97
12.4 Running the CV Tap/Delay Test . . . 97
12.4.1 Handling Failure . . . 97
12.5 Running the Dynamic Range Test . . . 97
12.5.1 Handling Failure . . . 98
12.6 Running the RF Leakage Test . . . 98
12.6.1 Handling Failure . . . 98
12.7 Running the ADC Noise Test . . . 98
12.7.1 Handling Failure . . . 98
12.8 Manually Testing the System Noise Floor Using a Spectrum Analyzer . . . 99

12.9 Running the 200 MHz 4 node Test . . . 99
12.9.1 Handling Failure . . . 99
12.10 Running the Rx Chain Test in Verify Mode . . . 100
12.10.1 Handling Failure . . . 100
12.11 Running the RFDB Loopback Test in Verify Mode . . . 100
12.11.1 Handling Failure . . . 100
12.12 Running the Rx Chain Test in Calibrate Mode . . . 101
12.12.1 Handling Failure . . . 101
12.13 Running the RFDB Loopback Test in Calibrate Mode . . . 101
12.13.1 Handling Failure . . . 102
12.14 RF Chain Calibration . . . 102
12.14.1 How the calibration is used . . . 103

13 Externally Controlling the RF environment **103**

14 Troubleshooting/FAQ **104**
14.1 Select Resource Profile prompt appears or RFview does not communicate with the hardware 104
14.2 CEC/Internal model test . . . 104
14.3 Wireshark Plugin . . . 105

15 Acronyms **105**

1. Introduction

This manual describes the setup and operation of the RFnest hardware and software. The primary user interface (RFview Software), the Channel Emulation Controller (CEC), the RFnest hardware, and their associated capabilities are detailed in this document.

Note: Throughout this document, the convention used is to describe meaning and options of the widgets and software components that pertain to a particular section of the manual (if they have not already been described) under the first sub-section of that section, followed by more detailed recommendations and uses in subsequent sub-sections. Since RFnest is a complex system, the user should read and understand the majority of this document before expecting to use the system successfully.

For detailed information on RFnest APIs, please refer to the RFnest API document.

The documentation of CORE and EMANE can also be found from NRL's website, currently at:

EMANE User's Guide: `http://downloads.pf.itd.nrl.navy.mil/docs/emane/emane.pdf`

CORE User's Guide: `http://downloads.pf.itd.nrl.navy.mil/docs/core/core_manual.pdf`

Note that the above links are subject to change.

2. Prerequisites

2.1 Supporting Equipment

The following equipment is required:

- A modern PC/laptop running a supported Linux OS and software dependencies (see Section 2.2). Intel Core i7, 8 GB RAM, 10 GB disk space, and a 1920x1080 display are recommended. The system will operate with a less capable computer, but may be sluggish/crowded in more complex scenarios.
- **D-Series:** a Gigabit Ethernet switch (a 10/100 Mbit only switch is not acceptable)
- **A-Series:** a 10/100 Mbit Ethernet switch (a Gigabit switch is not acceptable)
- Ethernet cables (at least 2 for A-series, 6 for D-Series)
- **D-Series:** a 15A outlet
- **D-Series:** a climate controlled room, typical office temperatures are acceptable but the unit should not be used outdoors or in a room where temperatures exceed 75 degrees Fahrenheit.
- SMA male RF cables (1 per radio for A-Series, for D-series at least 2 per radio, 3 if using splitters)
- RF splitters (required if using radios with a single tx/rx port with the D-Series, or radios with separate tx/rx ports with the A-Series)
- Attenuators (misc sizes, typically 10-20 dB increments, as needed to bring the input signal level to within spec, and to lower the output power further if needed)
- Shielding (as needed to prevent radios from talking over the air, may not be needed depending on radio type. Various RF enclosures are commercially available)
- Spectrum analyzer (optional, for testing/verification)
- Signal generator (optional, for testing/verification)
- RF devices (radios or other RF equipment being evaluated by the system)

2.2 Linux OS set up / supporting software configuration instructions

The following software is required to run RFview and CEC on the same PC.

- Linux OS -- Red Hat Enterprise Linux 7/Fedora 19/Ubuntu 12.04 or higher
- Java 1.7.0 (note Java 1.6 and 1.8 are not currently supported)
- CORE 4.6
- EMANE 0.8.1.x
- Libxml2

The UDP receive and send socket buffer sizes need to be increased.

In linux, add this to /etc/sysctl.conf:

```
net.core.wmem_max = 12582912
net.core.rmem_max = 12582912
```

Then, run `sysctl -p` as root to immediately apply these settings to the running system without rebooting. This config file is automatically read and applied on future boots.

Under Linux, Java needs to be run using the option to prefer IPv4. This is handled by the rfview.sh script provided with RFview.

2.3 Windows OS Setup

RFview can be run under Windows. The CEC, however, requires EMANE, which is only available for Linux. Java 1.7 is the only requirement for RFview.

The following registry key should be set to adjust UDP buffer sizes in Windows.

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Afd\Parameters]
DefaultReceiveWindow = 12582912
DefaultSendWindow = 12582912
```

2.4 Java installation instructions

Oracle's Java is recommended and can be downloaded from `http://www.java.com`. Version 1.7 is required.

On Linux, OpenJDK can be used if normal Java is not available. OpenJDK may introduce problems, especially for older systems or very complex scenarios. If any problems are observed with RFview and OpenJDK is in use, it is recommended to switch to Oracle's Java implementation.

Installing OpenJDK on Red Hat/Fedora:

```
yum install java-1.7.0-openjdk-devel
```

Installing OpenJDK on Ubuntu/Debian:

```
apt-get install openjdk-7-jdk
```

2.5 CORE installation (OPTIONAL)

CORE is only required only for using virtual nodes in CORE, which is not the standard use case of RFnest. Most users can skip this step.

Get the Fedora (rpm) or Ubuntu (deb) package for your architecture (64-bit or 32-bit) from the following URL. Be sure to pick "all downloads" from the drop-down.

https://code.google.com/p/coreemu/downloads/list

Use your package manager to automatically install dependencies:

Red Hat/ Fedora:

```
yum localinstall core-daemon-4.6-1.fc19.x86_64.rpm
yum localinstall core-gui-4.6-1.fc19.noarch.rpm
```

Ubuntu/Debian:

```
software-center core-daemon_4.6-0ubuntu1_precise_amd64.deb
software-center core-gui_4.6-0ubuntu1_precise_all.deb
```

More information on CORE installation or how to install from source can be found at: https://code.google.com/p/coreemu/wiki/Quickstart

2.6 EMANE Installation:

If using Red Hat/Fedora dependencies are automatically resolved if you use `yum localinstall`. In Ubuntu, dependencies (all packages) need to be specifically installed:

Red Hat/ Fedora:

```
# install dependencies
sudo yum -y install openssl-devel perl-XML-Simple perl-XML-LibXML
# download and install EMANE 0.8.1-release-2
wget http://downloads.pf.itd.nrl.navy.mil/emane/archive/0.8.1-r2/
                emane-0.8.1-release-2.f19.x86_64.tgz
tar xzf emane-0.8.1-release-2.f19.x86_64.tgz
sudo yum localinstall emane-0.8.1-release-2/RPMS/F19/x86_64/*.rpm
```

Ubuntu/Debian:

```
# install dependencies
sudo apt-get install libssl-dev libxml-libxml-perl libxml-simple-perl
# download and install EMANE 0.8.1-release-2
wget \
http://downloads.pf.itd.nrl.navy.mil/emane/archive/0.8.1-r2/
                emane-0.8.1-release-2.ubuntu-12_04.amd64.tgz
tar xzf emane-0.8.1-release-2.ubuntu-12_04.amd64.tgz
sudo dpkg -i emane-0.8.1-release-2/deb/ubuntu-12_04/amd64/*.deb
```

More information on EMANE installation or how to install from source can be found at: https://code.google.com/p/coreemu/wiki/EMANE

2.7 Libxml2 install

Red Hat/Fedora:

```
sudo yum install libxml2-devel
```

Ubuntu/Debian:

```
sudo apt-get install libxml2-dev
```

3. RFnest Software Installation

3.1 RFview Installation

Unzip (windows) or un-tar (Linux) the RFview archive. In Mac OS, copy to the Applications folder. A `rfview.sh` command is provided for Linux users, and a `rfview.exe` is provided for Windows users. It is important to use these commands when running RFview so that the proper Java virtual machine settings are used.

3.2 CEC installation

Note that the CEC requires EMANE 0.8.1 to be installed. To install CEC, go to the cec directory and run `make` to compile the CEC. Next, run `make install` as root. This copies the resulting libcec.so file to the appropriate library path (typically `/usr/local/lib/`) so it can be used by EMANE. It may also be necessary to run ldconfig as root if the library cannot be found by EMANE. Sensible default settings for the CEC are provided in the cec.xml and eventdaemon.xml configuration files.

*g++ is required to compile CEC. To install g++:

Red Hat/Fedora

```
sudo yum install gcc-c++
```

Ubuntu/Debian

```
sudo apt-get install g++
```

4. CEC Configuration

The CEC acts as an intermediary between the RFnest™ hardware and other components (e.g. RFview), performing basic channel modeling, translating APIs and interacting with low-level hardware behaviors.

The CEC is configured via an EMANE-based XML.

Ensure that the analog parameter in cec.xml is set correctly or the system will not work correctly. Use `analog=on` when using analog RFnest™ and `analog=off` when using digital version. Also, in analog mode the calibration file should be set based on the frequency being used (ex. `frequency_calib=calib/2400.calib` for 2.4GHz), otherwise the system will not be calibrated.

Sample cec.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventagent SYSTEM "file:///usr/share/emane/dtd/eventagent.dtd">
<eventagent name="Channel Emulator Controller" library="cec">
  <param name="publishpathlossevents" value="off"/>
  <param name="publishlocationevents" value="off"/>
```

```
  <param name="processpathlossevents" value="off"/>
  <param name="processlocationevents" value="on"/>
  <param name="basicmodel" value="on"/>
  <param name="analog" value="off"/>
  <param name="emanetestmode" value="off"/>
  <param name="testmode" value="on"/>
  <param name="testcount" value="20"/>
  <param name="testincrement" value="0.0001"/>
  <param name="freespacemodel" value="on"/>
  <param name="freespacewavelength" value="0.125"/>
  <param name="tworaymodel" value="off"/>
  <param name="longdistpathlossmodel" value="off"/>
  <param name="longdistpathlossexponent" value="2.0"/>
  <param name="hatamodelurban" value="off"/>
  <param name="hatamodelsuburban" value="off"/>
  <param name="hatamodelrural" value="off"/>
  <param name="hatapcsmodelurban" value="off"/>
  <param name="hatapcsmodelsuburban" value="off"/>
  <param name="hatafrequency" value="2400"/>
  <param name="fadingmodel" value="off"/>
  <param name="frequency_calib" value="/home/username/cec/calib/2400.calib"/>
  <param name="antennapattern" value="off"/>
  <param name="antennaprofile" value="/home/username/cec/antenna/antenna2.xml"/>
  <param name="dopplerincrement" value="7.45058097"/>
  <param name="manual_control" value="off"/>
  <param name="number_of_nodes" value="12"/>
</eventagent>
```

5. Starting the System

5.1 Basic Cabling (A-Series)

The A-Series hardware must be connected to via 10/100 Ethernet (**NOT** Gigabit Ethernet). Failure to use a 10/100 only (non-Gigabit) switch between the control PC and the hardware will prevent the hardware from being controlled properly. No USB connections are required for A-Series.

Connect the system's supplied 5V power adapter and turn on the power switch in the rear of the unit. The front green light will illuminate when the A-Series unit is receiving power.

5.2 Basic Cabling (D-Series)

Before powering on the system, the power and Ethernet cables need to be connected properly.

Ethernet: All hardware boards (the CEB and all DDBs) must be connected to a Gigabit Ethernet switch. Ethernet should be connected to the front left Ethernet jack on the front of the unit on the lowest board (CEB). Ethernet must also be connected to the rear connector for each DDB. A D512 system should have a total of 4 Ethernet connections, one to the front connector of the CEB and 3 to the rear connector of the DDBs. Note that if firmware revision 1 is being used (i.e. RFview 2.11), the rear Ethernet connector on the CEB is no longer functional, and the Ethernet cable must be switched to the front left CEB connector. The PC(s) running the CEC and RFview must be directly connected to this switch as well. The PC(s) running CORE/EMANE and any other data collection tools must be connected to this switch or have appropriate unicast and multicast routing setup to enable them to get data.

Power: The system must be connected to a 110v outlet capable of supplying 5A per 24 node module present in the system, in addition to any power load caused by other components (PCs, radios, frequency converters, etc.).

USB (optional): If a firmware upgrade is provided, connect a micro USB connector to the JTAG port on the boards receiving the new firmware. The JTAG port is on the front for the CEB and the rear for each DDB. Be **EXTREMELY** careful to orient the micro USB plug correctly. Forcing it in while oriented incorrectly will damage the board. In general, the USB connector should NOT be used except while an upgrade is taking place, and should be disconnected after the upgrade process completes.

5.3 Resource Profiles (D-Series Only)

The following resource profiles are available by default. Others are possible. The resource profile must be selected and loaded before a scenario is created and the hardware is used. A scenario created using one resource profile is incompatible with another resource profile.

ID	Chassis	Ports	Sample Rate (Hz I&Q)	Multipath Taps	Per-tap Doppler	Buffer Length	Doppler (Hz) Resolution[1]	Doppler (Hz) Increment[1]
1	Large	12	250	4	No	256000	0.465661287	0.058207661
2	Large	24	31.25	20	No	256000	0.465661287	0.007275958
3	Large	8	250	8	No	256000	0.465661287	0.058207661
4	Small	12[2]	312.5	4	Yes[3]	64000	0.291038305	0.072759576
5	Small	12	250	4	Yes[3]	256000	0.465661287	0.058207661
6[3]	Small	12	125	5	Yes[3]	256000	0.465661287	0.02910383
7[3]	Small	12	62.5	13	Yes[3]	256000	0.465661287	0.014551915

ID	Time Resolution (ns)	Long Delay	Maximum Long Delay (ns)	Per-channel Delay	Maximum Per-channel Delay (ns)	Maximum Per-tap Delay (ns)
1	4	No	n/a	No	n/a	4,000
2	32	Yes	2,000,000,000	No	n/a	32,000
3	4	No	n/a	No	n/a	4,000
4	3.2	No	n/a	Yes[3]	25,600[3]	3,200
5	4	No	n/a	Yes[3]	32,000[3]	4,000
6[3]	8	No	n/a	Yes	360,000	8,000
7[3]	16	No	n/a	Yes	720,000	16,000

[1] The Doppler resolution is the precision with which the value is controllable (whether via RFview or via the API). The Doppler increment is a constant that is used when **formatting** the API messages sent to the RFnest hardware.
[2] Profile 4 uses a hardware configuration with 16 ports, however ports 3, 7, 11, and 15 are disabled making 12 active ports.
[3] **Planned capability, not currently available**

5.4 Network Startup

The software requires a correctly configured Ethernet interface and multicast route **prior** to the CEC and RFview being run. Note that if RFview is being used in replay mode only, this step is not necessary.

Configure the Ethernet interface (assuming eth0 is the PC's Ethernet interface). Under Ubuntu this may instead be p5p1 and under RHEL7 this may be enp14s0.

```
ifconfig eth0 10.5.10.200/24 up
```

Configure the multicast route:

```
ip route add 224.0.0.0/4 dev eth0
```

NOTE: The hardware always sends packets from the 10.5.10.0/24 subnet. If the PC lacks a valid route for this subnet that is directed to the PC's Ethernet port, the OS's anti-spoofing logic may drop all packets from the hardware. This can be easily accomplished by giving the PC an IP in this subnet (e.g. 10.5.10.200 as described above) or by adding a secondary IP to the PC that is in that subnet as follows if the user does not want to change the PC's primary IP.

```
ip route add 10.5.10.200/24 dev eth0
```

The system firewall must be configured to allow data to be sent into the PC. These ports are 45703 (the default EMANE port), 20010, and the range 20850-20870. The following commands can be run in RHEL7 to accomplish this, and must be run at each boot of the system. The firewall_setup.sh script in the cec directory contains these commands and can be run in their place.

```
firewall-cmd --direct --add-rule ipv4 filter IN_public_allow 0 -m udp -p \
  udp --dport 45703 -j ACCEPT
firewall-cmd --direct --add-rule ipv4 filter IN_public_allow 0 -m udp -p \
  udp --dport 20010 -j ACCEPT
firewall-cmd --direct --add-rule ipv4 filter IN_public_allow 0 -m udp -p \
  udp --dport 20850:20870 --j ACCEPT
```

5.5 Running the CEC

To start the CEC, run the script `run.sh` in the `rfnest_cec` directory. This invokes the emaneeventd process with the appropriate settings. The CEC should be run prior to RFview or other third party software, otherwise the initial positions for stationary nodes in the scenario might not be received by the CEC when the scenario is started.

5.6 RFview User Interface Startup

To start the RFview user interface on Windows, double click the rfview.exe file. Mac OS X users should drag the RFview bundle to their Applications folder.

Windows users should use the rfview.exe executable.

Linux users should use the shell script rfview.sh.

```
./rfview.sh
```

To manually launch the RFview application type the following command:

```
java -jar rfview.jar
```

RFview can also be launching using a manual classpath using the following command:

```
java -cp <classpath> com.iai.rfnest.workbench.AppMain
```

5.7 Selecting a Resource Profile (D-Series Only)

When RFview is started, a prompt may appear indicating the hardware was not detected and asking the user to select a resource profile. If RFview is being run without the hardware (e.g. to define a scenario, analyze recorded data from a prior run, demonstrate a recorded run, etc.) the user should select the resource profile that was or will be used to create the scenario. RFview will then operate as if the hardware is configured with the chosen resource profile. If the user is intending to run RFview with the hardware, the presence of this prompt is an indication that RFview did not receive a response when the hardware was queried. Check that all boards have only their green LED lit on the front panel. If this is the case, then the network connectivity between the hardware and the PC, or the multicast route on the PC, is not correct.

5.8 Verifying System Operation

Once RFview has started, the following demonstrate the system is fully operational:

1. RFview does NOT prompt the user to select a resource profile (unless the hardware is not intended to be used).
2. **D-Series Only:** The bottom status bar in RFview indicates a blue circle for the CEB and all DDBs present. A D512 system for example will have 3 DDBs present and 1 CEB.
3. A scenario can be defined/loaded and the "Initialize" button completes, enabling the play button.
4. **D-Series Only:** A basic digital built-in test passes on the first or second try, for example the Dynamic Range test. The Tap/Delay test can also be used, however it is more prone to false negatives.
5. Use the Channel Properties Editor (see Section 8.12) to control a channel and observe the expected signal change via a radio, spectrum analyzer, or other device that reports the power level.

6. Optional Configuration

The following instructions are optional, and do not apply to all systems or situations. For the first use of the system, it is recommended to skip to Section 7.

6.1 Compiling RFview

Compiling RFview is optional. A cross-platform Java archive is already provided and ready for immediate use; RFview source code is typically not provided. RFview uses ANT scripts for compilation. These scripts are executed:

1. Compiling RFview
   To compile all the RFview modules, and generate the distribution for all platforms, execute the following commands:

   ```
   cd utilities
   ant clean build all
   ```

2. Individual modules
   All RFview modules contain a series of ANT scripts that compile the project, run the unit tests, generate the Javadocs, and generate the code quality reports. To build a module, type the following command:

   ```
   ant build
   ```

To build, run the unit tests, generate the Javadocs, and generate the code quality reports, type the following command:

```
ant all
```

3. Generate Sources
   The RFnest Core Module contains an auto-generated XML parser which needs to be manually generated whenever the object model is modified. To generate source files, type the following command:

```
ant generate-sources
```

6.2 Optional Impact and Driver Installation (if needed)

If the RFnest hardware does not automatically display a green light at each board in the system approximately 30 seconds after being powered on, this section applies. For nearly all systems, the green lights come on automatically and this section can be skipped. If a firmware upgrade was provided, Lab Tools must be installed in order to perform the upgrade.

For RFnest hardware which must be loaded via USB, Xilinx Lab Tools must be installed to allow programming of the FPGAs via USB. These tools along with installation (for Windows and Linux) are available from Xilinx.

6.3 RFview Configuration

RFview has several configuration files. Sensible defaults are provided for most situations.

rfview.properties is used to define the Java System properties of the RFview application. The contents of this file are loaded into the System Properties as the first line of the Main Program. This file is used to setup any system parameters as well as setting the preferred base map.

hardware_props.xml is used to determine the acceptable RFnest hardware unit this software is configured to communicate with. Information stored within this file contains, which hardware series Analog or Digital, the number of ports, the number of physical ports, the number of virtual ports, the port configuration per group, and numerous other parameters. The RFnest team is responsible for maintain and packaging the Hardware Properties file for RFnest users.

*All supported hardware properties are stored in the `hardware_properties` folder.

6.4 RFview Map Configuration

RFview supports a configurable base map, powered by IAI's QuickMaps library. The base map is the background map for all RFview scenarios for a given software installation. To change the base map for RFnest, modify the rfview.properties at the root directory of the RFview installation.

The `com.iai.rfview.baseMap.source` property indicates the QuickMaps map source. Map sources indicate where the map data comes from. Valid values are openstreet and virtualearth. The `com.iai.rfview.baseMap.type` property indicates the QuickMaps map type. Map types indicate what type of map data to get from the map source. Not all map sources support all map types. Valid values are street, terrain, and satellite.

The following rfview.properties configuration file contains various valid base map configurations. Uncomment the desired setting.

```
# This file contains the system properties for the RFview application.

# The default map configuration is:
com.iai.rfview.baseMap.source=openstreet
com.iai.rfview.baseMap.type=street

# For Bing Street maps
# com.iai.rfview.baseMap.source=virtualearth
# com.iai.rfview.baseMap.type=street

# For Bing Satellite
# com.iai.rfview.baseMap.source=virtualearth
# com.iai.rfview.baseMap.type=satellite

# For Bing Terrain
# com.iai.rfview.baseMap.source=virtualearth
# com.iai.rfview.baseMap.type=terrain}

# This line is required to correctly configure Java on dual IPv4/IPv6 stack PCs
java.net.preferIPv4Stack=true
```

The default base map property can also be assigned at compile time. This is useful for generating distributions that are automatically configured to use specific alternate map sources.

```
cd utilities
ant clean build all -DbaseMap.source=<source> -DbaseMap.type=<type>
```

6.5 Pre-caching RFview Map Data

The shell script `rfnest-mapcache.bat` on Windows and `rfnest-mapcache.sh` on Linux and Mac OS X are used to precache map data for the base map of the RFview workbench. It requires four command-line arguments specifying the geographic region where data should be acquired.

The script is powered by QuickMaps Cache Downloader. For more details, consult the com.iai.quickmaps.tiles.cache project page.

To cache map data for Melbourne, FL., Windows users should run:

```
rfnest-mapcache.bat -80.739288 28.231339 -80.540848 27.976346
```

To cache map data for Melbourne, FL., Linux users should run:

```
rfnest-mapcache.sh -80.739288 28.231339 -80.540848 27.976346
```

6.6 Generating Tactical Graphics for use in RFview

RFview comes prepackaged with a small set of MIL-STD 2525 tactical graphics. Users can generate their own tactical graphics via the shell script `rfnest-milstd2525-update.bat` on Windows and `rfnest-milstd2525-update.sh` on Linux and Mac OS X. It simply requires an active Internet connection, a valid SIDC Code, and an output file name. Consult the MIL STD 2525C specification for individual SIDC codes.

Under the hood this process is powered by NASA WorldWind.

To create MIL-STD 2525 tactical graphic using the SIDC Code (`SFAPMFF---*****`), Windows users should:

```
rfnest-milstd2525-update.bat SFAPMFF---***** Fighter
```

To create MIL-STD 2525 tactical graphic using the SIDC Code (`SFAPMFF---*****`), Linux users should:

```
rfnest-milstd2525-update.sh SFAPMFF---***** Fighter
```

6.7 Additional CEC Standard Mode Parameters

The following parameters control the CEC in normal operation.

analog if value=on, CEC is operating in analog mode. It is important to set this to be on for the software to function properly with A-series RFnest hardware and off for the software to function properly with D-series RFnest hardware.

frequency_calib value is path to calibration file. Applicable for analog mode only. It is important to set this file correctly relative to the frequency used by the radios to achieve the best performance

publishpathlossevents if value=on, EMANE pathloss events are published.

publishlocationevents if value=on, EMANE location events are published.

processpathlossevents if value=on, EMANE location events are processed.

processlocationevents if value=on, EMANE location events are processed.

basicmodel if value=on, use built-in models. If value=off, the user must supply their own software to repond to CCR Request API messages.

6.8 CEC Test mode parameters

The following are test mode model settings. In normal operation, these settings are ignored and cec uses the models and settings configured in RFview.

testmode if value=on, sets the CEC to run debugging tests for pathloss calculations.

testcount the number of distances to test if testmode is on.

testincrement the test distance increment in degrees longitude (0.0001 is approximately 10m).

emanetestmode if value=on, sets CEC to save debugging info from EMANE updates.

freespacemodel if value=on, free space model is used to calculate pathloss.

tworaymodel if value=on, two ray model is used to calculate pathloss.

logdistpathlossmodel if value=on, log distance model is used to calculate pathloss.

hatamodelurban if value=on, hata model for urban communication is used to calculate pathloss.

hatamodelsuburban if value=on, hata model for suburban communication is used to calculate pathloss.

hatamodelrural if value=on, hata model for rural communication is used to calculate pathloss.

hatapcsurban if value=on, hata PCS model for urban communication is used to calculate pathloss.

hatapcssuburban if value=on, hata PCS model for suburban communication is used to calculate pathloss.

fadingmodel if value=on, fading is applied to gain/pathloss values.

freespacewaveelength wavelength used by free space model in test mode only (e.g. 0.125 for 2.4 GHz).

logdistpathlossexponent exponent used by log distance model in test mode.

hatafrequency frequency used by hata models in test mode.

dopplerincrement this is a doppler correction factor used during normal operation.

antennapattern if value=on, sets CEC to use antenna pattern profile for radios.

antennaprofile value is path to an antenna pattern file, used for testing.

manual_control if value=on, the CEC uses the number_of_nodes parameter rather than RFview-provided messages.

number_of_nodes the number of nodes in the RFnest hardware, only applies if manual_control=on.

6.9 CEC Eventdaemon.xml Configuration

The eventdaemon.xml file is used to configure the EMANE infrastructure to match the network settings and the settings of other EMANE scenarios being integrated with RFnest.

eventservicegroup the group address for EMANE events.

eventservicedevice the device that should be listening to EMANE events. **This parameter is optional, if not supplied, the route to the multicast IP specified in eventservicegroup is used to determine the interface.**

Sample eventdaemon.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE eventdaemon SYSTEM "file:///usr/share/emane/dtd/eventdaemon.dtd↩
    ">
<eventdaemon name="EMANE Event Service Daemon" nemid="1">
  <param name="eventservicegroup" value="224.1.2.8:45703"/>
  <param name="eventservicedevice" value="eth0"/>
</eventdaemon>}
```

6.10 Loading the Hardware (D-Series Only)

This step applies only to hardware which is not automatically configured. If the green light on each hardware board becomes illunimated within 30 seconds of powering on the hardware, ignore this entire section.

All DDBs must be connected to the PC via the micro USB connectors on their rear panels. The CEB must be connected to a PC via the micro USB connector on the CEB's **front** panel. This is a total of 7 USB connections for each 24 node module. It is recommended that a USB hub be used to connect all USB connectors simultaneously. Note that the USB connectors are not required after the hardware has been powered on and the firmware has been loaded.

To load the firmware into the hardware via USB, the impact utility is used. In the fpga directory, run the script `load_ceb_XXX.sh` for the desired CEB resource profile. For the DDBs, the same `load_all_ddbs.sh` script is always used. Note that other scripts may be provided for upgrading your system or other purposes, please refer to the specific instructions given with the firmware being loaded. These scripts each take a few minutes to run. The orange LED will light on each board as it is being programmed or when it is not programmed. When the board is correctly programmed, only the green LED will be active.

When the CEB has been loaded, it is ready to use and will be detected by RFview. The DDBs however, require the Set Resource Profile message to be sent to correctly configure their sampling rates and DDB IDs. This message is automatically sent when the Initialize button in RFview is pressed.

7. Basic Concepts

This section provides an overview of the system components, typical workflow, high-level concept used through RFview, best practices, and a single reference for where major functions/features can be found. Again, RFnest is a complex system which requires careful planning and verification to use successfully. Understanding this section and the features used by your scenario is critical to success and to avoiding struggles which may otherwise occur. Note that many features are provided to automate and simplify common issues faced when setting up complex scenarios, connecting radios, debugging waveforms, doing analysis, and demonstrating your findings.

7.1 Major System Components

The system consists of four major components 1) the RFview GUI which allows scenario definition, modeling, record/replay and analysis; 2) the Channel Emulation Controller (CEC) which coordinates the changes to the scenario over time; 3) the channel models, a default version of which are built-into and enabled by default within the CEC; and 4) the RFnest hardware which digitizes RF signals, applies channel effects, digitally creates RF signals, and converts the resulting combined channel environment back to an analog RF signal for all connected radios in real-time. Each of these components work together during the initialization and running of the scenario.

7.2 Typical Workflow

There are two approaches to creating and running a scenario. One tempting method is to build the scenario all at once and fine-tune things over many iterations of running the scenario. The other is to verify individual radios operate correctly in the system and gradually build more complex scenarios while preserving the correctness and the desired scenario attributes. Due to the complexity and the subtleties in which various aspects of the scenarios, radios, and network protocols interact, the latter approach is almost always much more efficient and practical while also allowing the user to get a subset of the desired scenario up and running almost immediately.

To build a scenario in this way, the best bet is to start with an example scenario or a previously created scenario. At a high level, the process is to start by deciding what the overall evaluation scenario will be and planning how many radios will be connected and which radios communicate within the same band. After that is complete, define a few platforms and their radios in RFview and verifying the operation of a single link. If importing from a supported third party application (e.g. AWSIM), do so now. Don't worry about positions, routes, or more complex scenario attributes yet, simply decide which radios belong at which platforms as you add radios. Next, connect and verify the operation of more radios of the same type. Now add two radios of the next type of radio followed by the rest of that type, and so on until all radios are present in the scenario. Now, adjust the Gain vs. Distance Overlay to match the radios' behaviors. These steps are described in great detail in Sections 8.1 through 8.4 and especially Section 9.

Once that step is complete, use the Gain vs. Distance Overlay and the Link Quality Overlay (with Expected gain enabled) to position nodes and form routes. The scenario can then be played (at 10-100x normal speed) to visualize what the **expected (but not guaranteed, otherwise this entire exercise would be pointless!)** behavior of links will be under this mobility. The status of links can be further verified by running the scenario at 1-10X speed and/or pausing the scenario at key points to verify whether radios' link statuses match those expected. If the radios or other parts of the system under test can measure their RSSI, this can be reported and displayed in RFview in real time to further streamline this process using this higher confidence information. These steps are described in Sections 9.8, 8.15, and 10.1.

Once the scenario is created, additional monitoring should be added. These include defining alerts, collecting and reporting routing paths (if available), and collecting and reporting various metrics of interest.

The entire system under test can now be verified in the scenario. This includes verifying multi-hop routing, interactions between multiple links, application layer performance, etc. These steps are described in Sections 9.3, 10.2 and 10.3.

Next, advanced concepts can be added including events, recording and replaying signals, injecting noise/EW signals (either digitally or by attaching a physical emitter device), adding EMANE-based software simulated/emulated radios or defining custom antenna patterns and delay profiles. These steps are described in Sections 11.1 through 11.7.

Finally, analysis can be performed using the record/replay functionality (Section 8.15), swimlanes (which capture each radio's transmit/receive/idle state and power level vs time) (Section 11.3), and exporting the recorded results, called the Event Deck, to external tools (Section 11.7).

7.3 Platform Types and Radio Types

RFview allows scenarios to be defined around the concept of platforms and radios. A platform is a physical entity that moves as a single unit. Typical platforms include a person, a vehicle, an aircraft, etc. Radios are real devices (or virtual devices in EMANE) such as RF transceivers (e.g., a cell phone, a WiFi radio, a tactical radio, etc.), emitters, listening devices, or anything else which will should transmit or receive RF signals as part of the scenario. Each platform can have one or more radios associated to it.

Be aware of the following caveats when assigning radios to platforms:

- Platforms are only for defining mobility and aiding the user by synchronously and logically moving multiple radios together. Platforms are not otherwise modeled. A platform with no radios serves only as a visual marker on the map and has no modeling, channel, or evaluation related impact.
- Platforms are modeled as a center of mass point in space. The radios attached to a platform may have different heights or antenna patterns, but are otherwise in the exact same position and so have very strong channels between each other (assuming they are in the same frequency group).
- If a single physical radio has multiple transceivers, meaning the radio can transmit/receive at multiple frequencies simultaneously, is full duplex (simultaneous transmit/receive), or operates in a large discontinuous band greater than the resource profile's bandwidth, this single physical radio must be defined in RFview as multiple radios. For example, a single physical radio with a uplink and downlink frequency which differ by approximately 500 MHz and/or are used simultaneously must be defined as two separate radios, one at each frequency.

Since a relatively small number of distinct types of platforms are commonly used, RFview allows the concept of "platform type" and "radio type" to create templates which are then easily instantiated multiple times within a scenario and simultaneously edited. Platform types and radio types can also be imported from an existing scenario when a new scenario is first created. Platform types and radio types must be defined prior to defining platforms and radios themselves, as all platforms and radios must be assigned a type. This definition is described further in Sections 8.1 and 8.2.

7.4 Groups and Frequency Groups

The sets of radios which may transmit/receive RF to/from each other are termed a "group." All radios must be members of exactly one group. Radios which are not part of the same group do not experience each other's RF via the system's hardware ports and in fact the attributes of channels between radios in different groups are not even calculated or definable in any way.

Groups can be either real or virtual. A real group represents a set of physical ports on the system hardware to which real RF devices can be connected. A virtual group represents a set of radios whose channel

attributes will be calculated and send via the system's APIs, for example to drive an EMANE-based virtual node model. Typically RFview is used with only the real groups defined by the current resource profile, however RFview may be used with both real and virtual groups simultaneously, or only with virtual groups.

Groups can also be arbitrarily subdivided into "frequency groups." A frequency group is a subset of a group's radios which share a common frequency. This allows flexibility in the use of radios operating at heterogeneous frequencies. Frequency groups are defined simply by assigning radios which are defined with different center frequencies in RFview to the same group. Note that frequency groups are logically treated as being in different groups; the channels between radios in different frequency groups, while technically present, are always set to infinite loss even if the frequency only differs by 1 MHz. This prevents otherwise erroneous mixing of signals at different frequencies within the hardware.

7.5 Maps and Overlays

RFview provides a geospatial view of the scenario and associated data. A variety of map sources can be used, including pre-cached maps for offline use. On top of these maps, platform positions, routes, links, and various collected and analysis-oriented metrics can be displayed. The Overlays tab provides methods for enabling/disabling each layer of this data, configuring the order of the layers, and editing the configuration for each layer.

7.6 Scenario Analysis, Recording, and Demonstration

RFview consists of not only tools to define and execute scenarios but to analyze and demonstrate the operation of the system under test either in real time or by recording the results of the scenario and then re-visualizing the recorded results. RFview provides time-synchronized geospatial and graph-based displays of both the scenario state (e.g., channel conditions, platform positions, etc.) and the behavior and outcome of the scenario (observed link states, arbitrary numeric metrics, and indicators of the paths taken by packets or other data through the scenario). To make maximum use of these capabilities, the user must configure the system under test or supporting test software and scripts to monitor the system under test and report this information. This allows the recorded scenarios to contain as much of the relevant data as possible, reducing the need to re-setup and re-run scenarios for debugging/analysis and providing maximum impact and understanding during demos. A scenario can be run multiple times and each run can be independently recorded. The recordings also be exported into a human-readable text form for more detailed analysis and importing into other tools.

8. Creating the Basic Scenario Attributes

This section describes how to define basic elements of the scenario including defining platforms, radios, building a dynamic scenario, and using the load/save and record/replay functions of RFview.

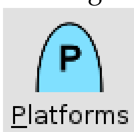
> *NOTE: After defining the platfoms, radios, and group configuration, prior to fine-tuning the scenario and creating detailed routes it is likely useful to skip forward to Section 9 and in particular Sections 9.5 through 9.7 which cover properly attaching radios, followed especially by Section 9.8 which provides key tools that greatly simplify the creation and understanding of the relative positions and mobility of platforms in the scenario.*

8.1 Platform Type Editor

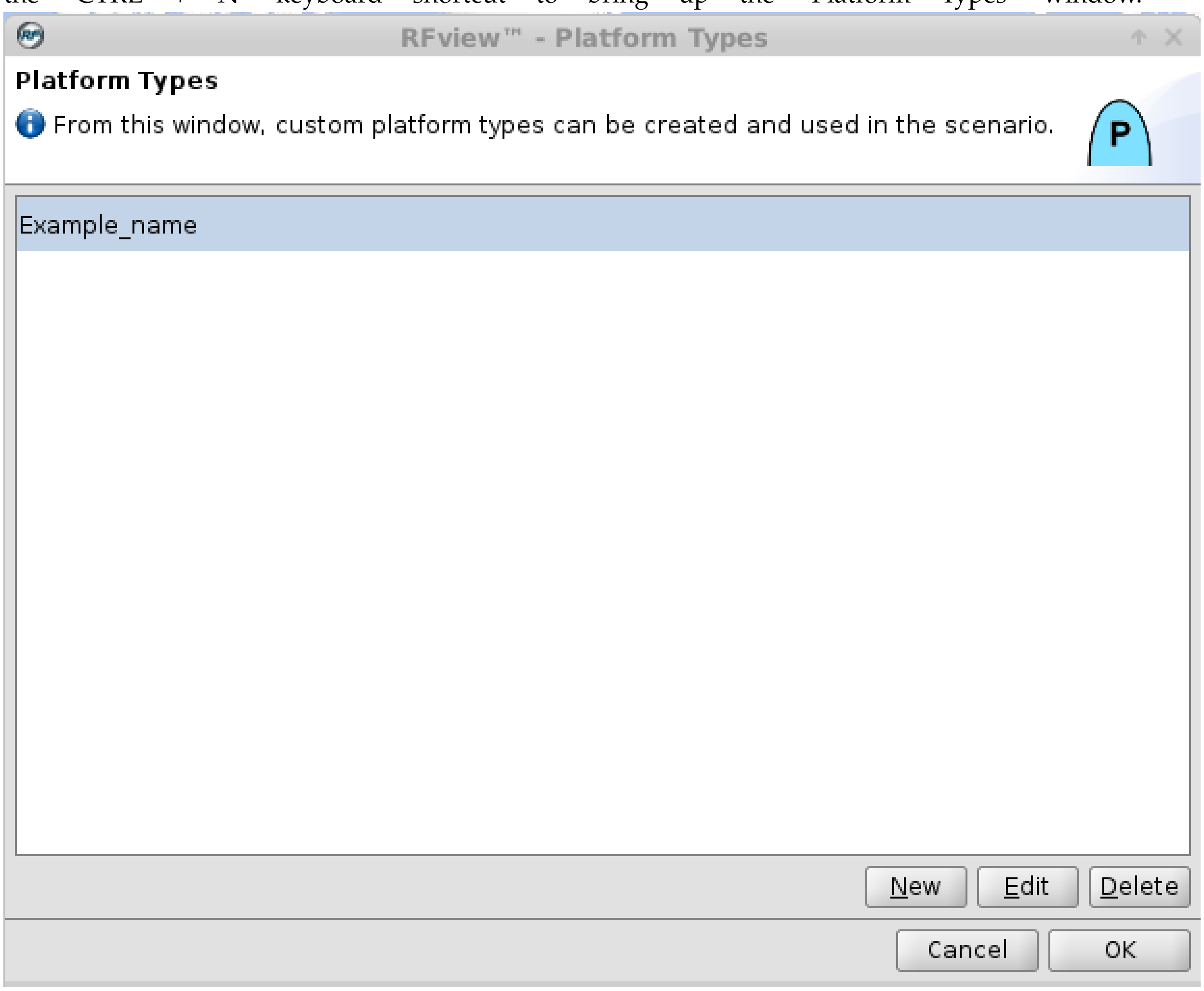
The Platform Type Editor allows the user to add and edit the type of nodes that will be placed into the scenarios. Platforms establish the icon and the mobility model for each physical node type. Typically users generate both air and ground platforms, (e.g., Dismounted Soldiers, Vehicles, Fast Moving Aircraft, Slow Moving Aircraft, Static Locations, etc.) Each node in the system needs an assigned platform type.

8.1.1 Create platform type

1. Open the platform type editor:
   (a) Clicking on 'Tools' and selecting 'Platforms' from the menu.
   (b) Clicking on 'Platforms' button on the main toolbar.



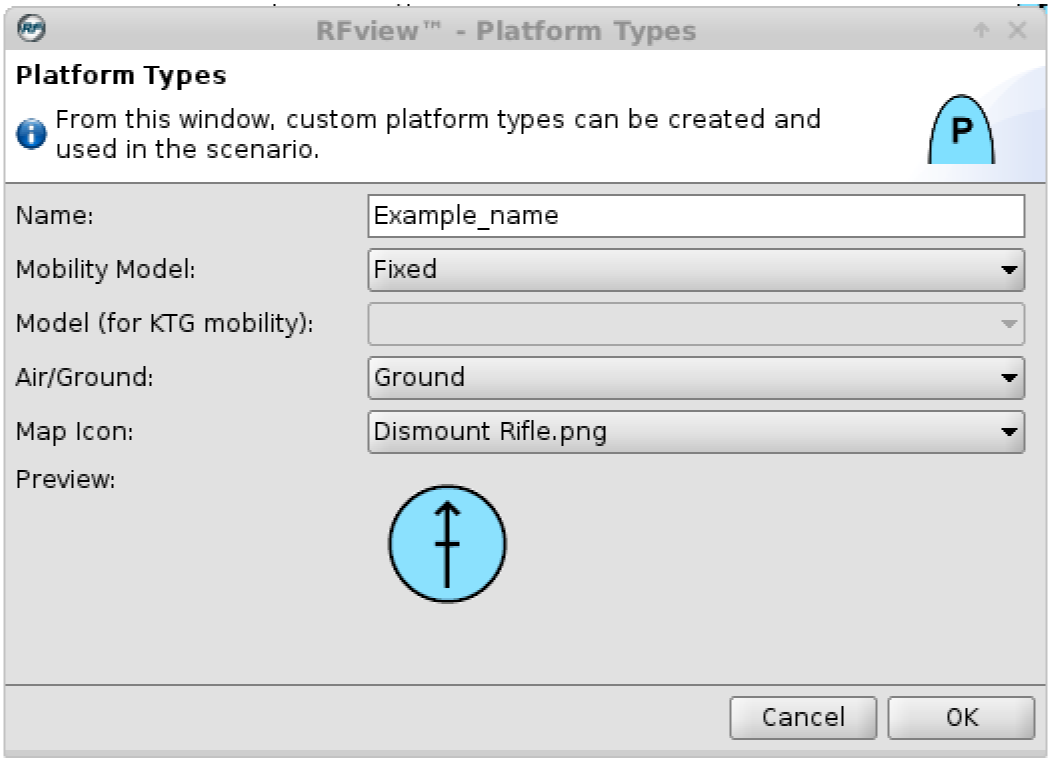
2. If no platform types exist the widget will automatically pop up a 'Platform Types' window.
3. If previously created platform type(s) exist, the user must click on 'New' or use the 'CTRL + N' keyboard shortcut to bring up the 'Platform Types' window.



4. Input platform type name in the 'Name' field.
5. Set mobility model from the 'Mobility Model' drop down menu.

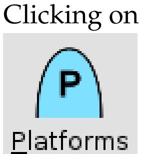
- If KTG mobility is chosen, user will also need to select a model for KTG mobility from the 'Model (for KTG mobility)' drop down menu. See Section 11.12 for more details on KTG.

6. Set the platform to be Air or Ground node using the 'Air/Ground' drop down menu.
   - If KTG mobility model is selected, the platform will automatically be set as an aerial node.
7. Set the icon for the platform type from the 'Map Icon' drop down menu.
8. Click 'OK' to save platform type.

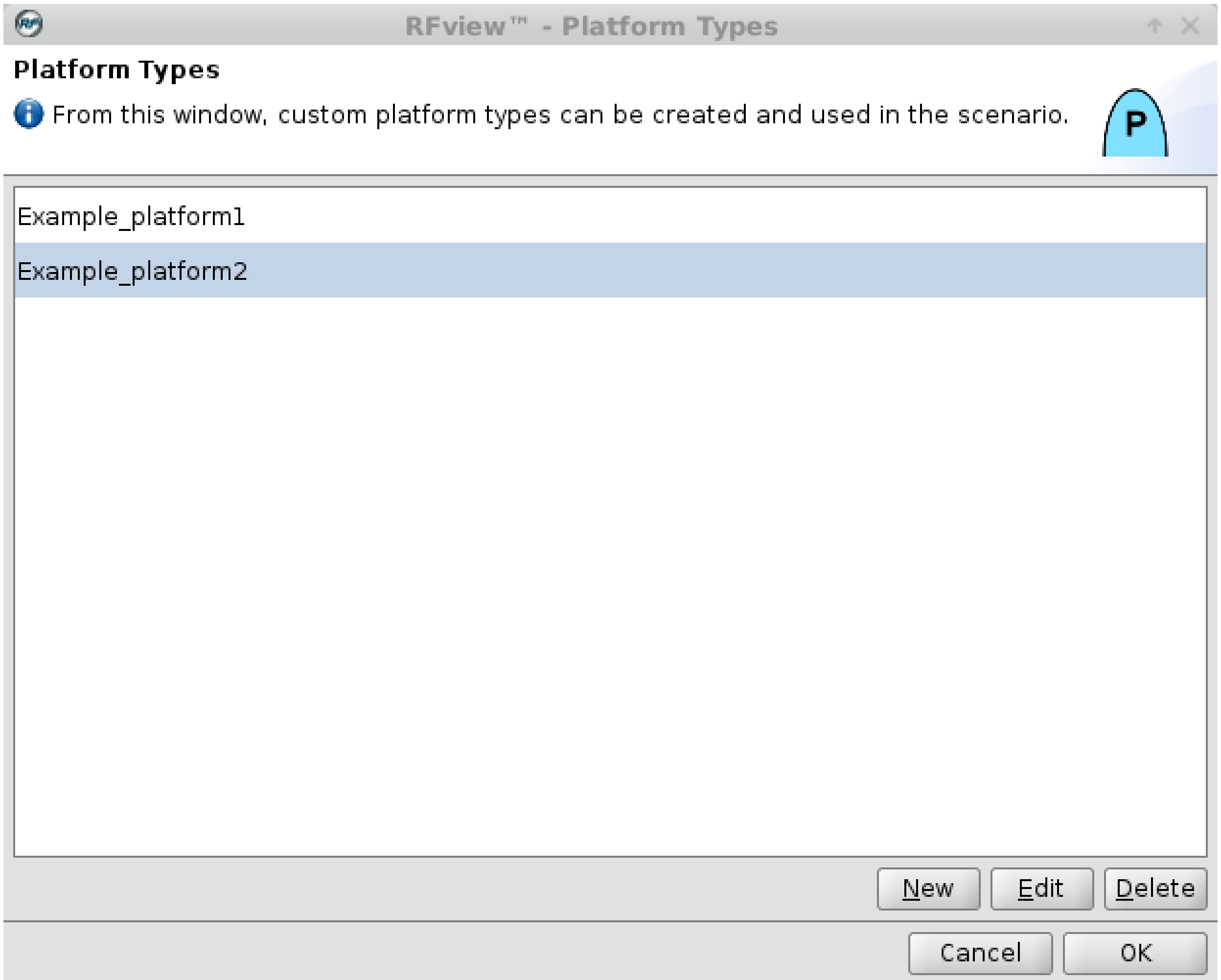


8.1.2 Edit platform type

1. Open the platform type editor:
   (a) Clicking on 'Tools' and selecting 'Platforms' from the menu.
   (b) Clicking on 'Platforms' button on the main toolbar.



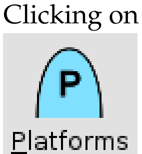
2. Select the platform type to edit from the list.



3. Click 'Edit' to bring up the 'Platform Types' window.
4. Edit desired fields.
5. Click 'OK' to save changes.

8.1.3 Delete platform type

1. Open the platform type editor:
   (a) Clicking on 'Tools' and selecting 'Platforms' from the menu.
   (b) Clicking on 'Platforms' button on the main toolbar.



2. Select the platform type to delete from the list.
3. Click 'Delete' to remove the platform.

8.2 Radio Type Editor

The Radio Type Editor allows the user to enter various radio types that will eventually be assigned to different nodes. The Radio Type Editor provides a single entry for radio settings that can be applied to multiple nodes. This is also the time when similar radios with different settings (frequency) should be set up as well.

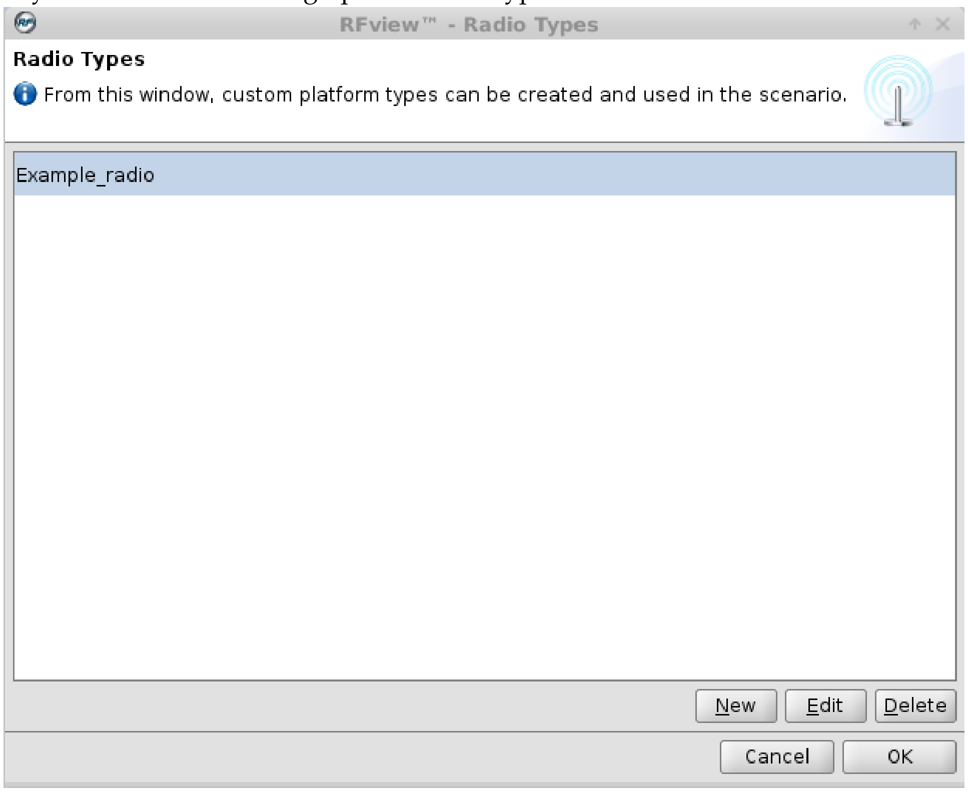
Note: If using the A-Series, radios could be set to any frequency within the specifications. The standard version is 1.8GHz to 2.8GHz, with best performance achieved from 1.8GHz to 2.5GHz. The frequency in this dialog is not applicable to the A-Series.

8.2.1 Create Radio Type

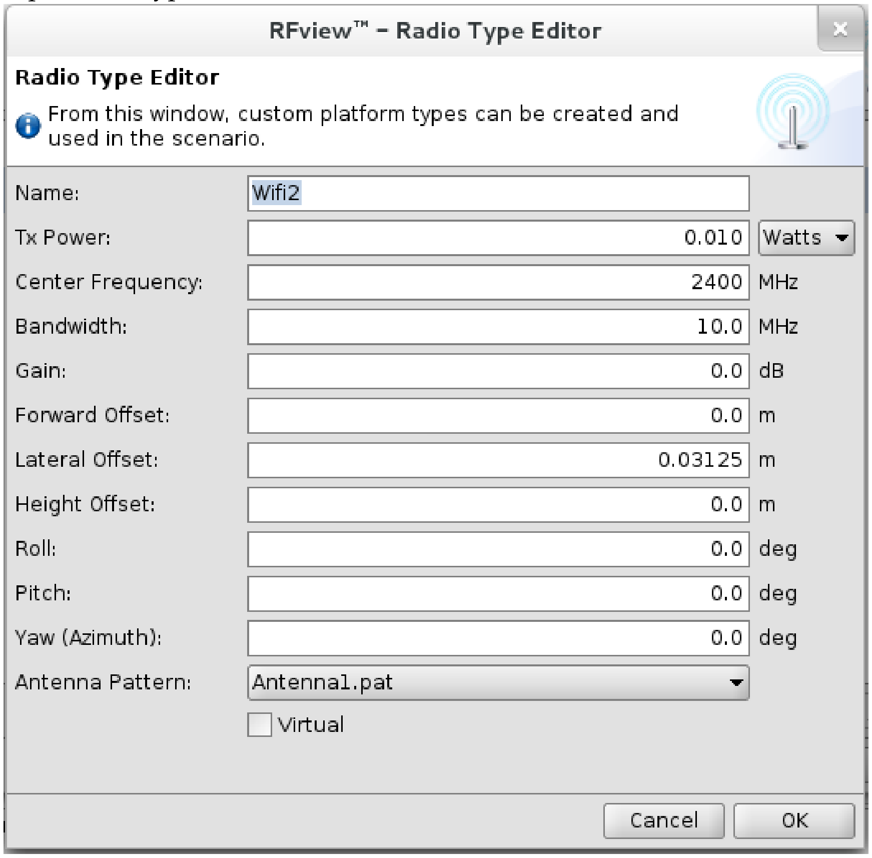
1. Open the radio type editor:
   (a) Clicking on 'Tools' and selecting 'Radios' from the menu.
   (b) Clicking on 'Radios' button on the main toolbar.

2. If no radio types exist the widget will automatically pop up a 'Radio Types' window.

3. If previously created radio type(s) exist, the user must click on 'New' or use the 'CTRL + N' keyboard shortcut to bring up the 'Radio Types' window.



4. Input radio type name in the 'Name' field.



5. Input transmission power in dBm in the 'Tx Power' field. This field acts as an offset for the Link Quality Overlay (see Section 8.9).

6. Input center frequency in the 'Center Frequency' field. This is the center frequency at which the RFnest hardware will sample the incoming RF signal, and does not necessarily have to be the center frequency of the radio. Ensure that the band used by the radio is inside the center frequency configured here. For the A-series RFnest, this field is not used.

7. Input bandwidth in the 'Bandwidth' field. This bandwidth is for visual reference. The RF bandwidth sampled is defined by the group in each resource profile and is not configurable. The radio's operating frequency plus radio's bandwidth must be within the RF center frequency and group bandwidth.

8. Input gain in dB in the 'Gain' field. This field is for visual reference.

9. The next six fields allow the user to input any offset information relative to the center point of the platform and relative to the platform's orientation – (leave the values as "0" if the antenna is effectively in the middle of the platform and pointed toward the nose of the platform as defined by the antenna pattern)

10. Input forward offset in meters in the 'Forward Offset' field. See section 8.2.4 for details.

11. Input lateral offset in meters in the 'Lateral Offset' field. See section 8.2.4 for details.

12. Input height offset in meters in the 'Height Offset' field. See section 8.2.4 for details.

13. Input antenna roll in degrees in the 'Roll' field. See section 8.2.4 for details.

14. Input antenna pitch in degrees in the 'Pitch' field. See section 8.2.4 for details.

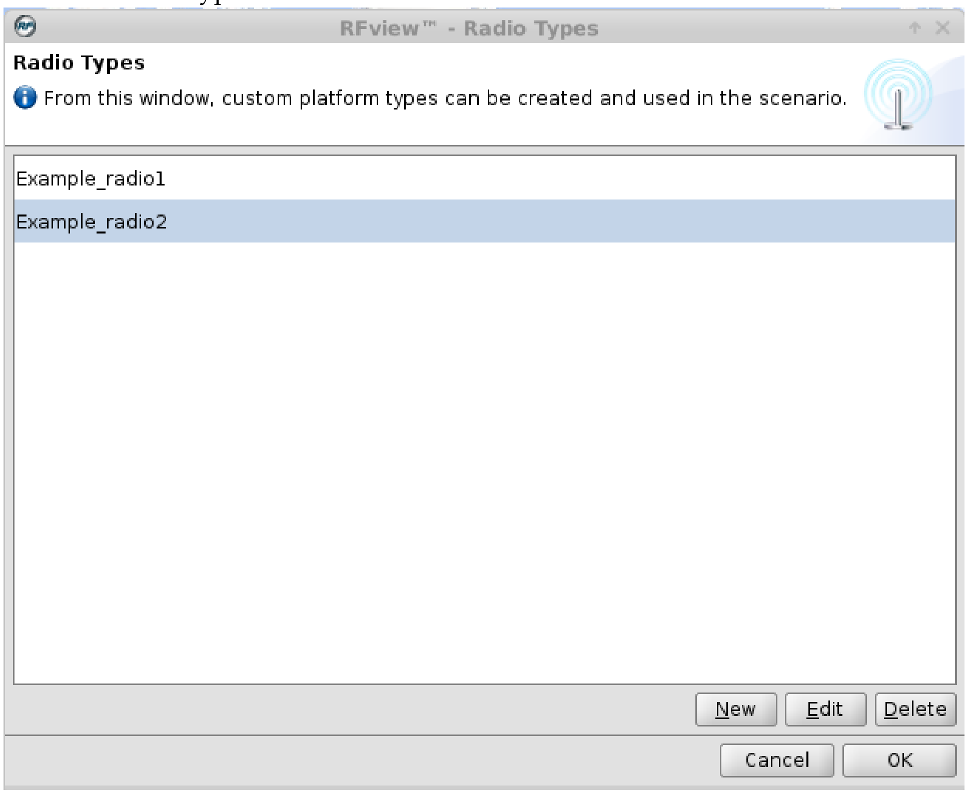
15. Input antenna yaw in degrees in the 'Yaw (Azimuth)' field. See section 8.2.4 for details.

16. Set antenna pattern from the 'Antenna Pattern' drop down menu. By default, the first Antenna pattern (Antenna1.pat) is used and is Omni-Directional. If there is a different 3D antenna pattern, the file should be selected.
17. Check the 'Virtual' checkbox if this radio type is virtual, if this is a real radio type leave the checkbox unchecked.
18. Click 'OK' to save radio type.

8.2.2 Edit radio

1. Open the radio type editor:
   (a) Clicking on 'Tools' and selecting 'Radios' from the menu
   (b) Clicking on 'Radios' button on the main toolbar.

2. Select the radio type to edit from the list.



3. Click 'Edit' to bring up the 'Radio Types' window.
4. Edit desired fields.
5. Click 'OK' to save changes.

8.2.3 Delete radio

1. Open the radio type editor:
   (a) Clicking on 'Tools' and selecting 'Radios' from the menu.
   (b) Clicking on 'Radios' button on the main toolbar.
   

2. Select the radio type to delete from the list.
3. Click 'Delete' to remove the radio type.

8.2.4 Radio Coordinate System

The coordinate system used for placing radios on platforms is "north, east, down" with the **exception that height offset is reversed**. Assuming the platform's roll, pitch, and heading are all 0, a positive forward offset means toward north, a positive lateral offset means toward east, and a positive height offset means up toward the sky. In other words, if the platform were a car, a positive forward offset is toward the hood, a positive lateral offset is toward the passenger side, and a positive height offset is toward the roof.

To determine the sign for orientation, the simplest method is to use your **right** hand as a reference. For roll, place your right hand so your right thumb is pointing forward. A positive roll rotates in the same way your wrist rotates if you bend your hand toward your arm (i.e. close your wrist); going back to the car the passenger side of the car is raised in this case. Similarly, a positive pitch is when your thumb is pointing to the right and your hand is bent toward your arm; the car's hood is raised. Finally, a positive yaw is when your thumb is pointing down (not up!) and your hand is bent toward your arm; the car rotates clockwise.

Note that radio offset and orientation is relative to the platform, i.e. radio orientations are **not** absolute. When both the radio and platform have non-trivial roll, pitch, and yaw, determining and applying the actual absolute orientation of the radio is done correctly by RFview but is not readily represented for the user.

Radio positions are maintained throughout the system in double-precision latitude, longitude, and altitude values, giving sub-centimeter precision for phase calculations. Note that delay is controllable with less precision, as specified in the resource profile specific delay resolution provided in Section 5.3 but generally with sub-meter precision.

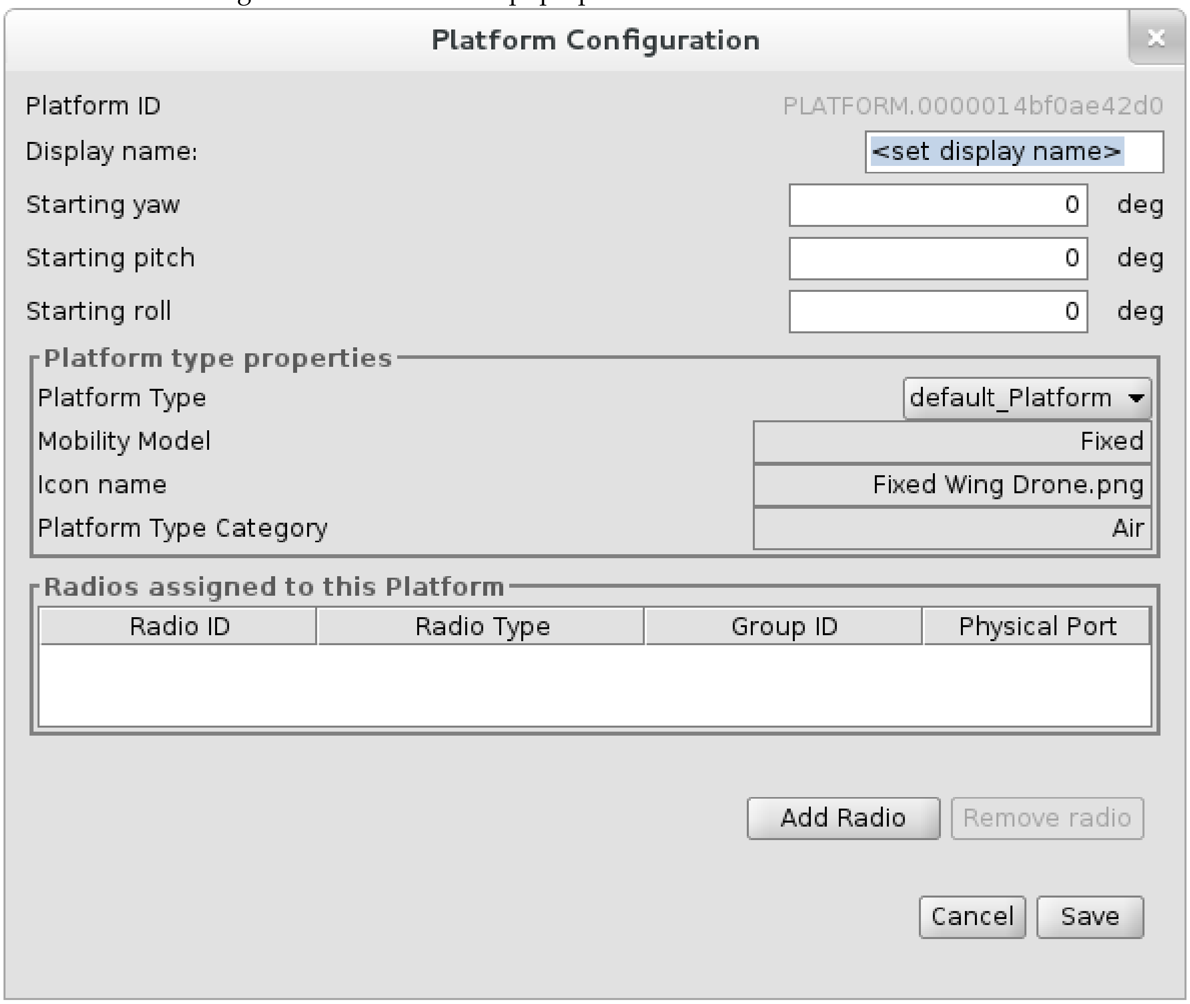
8.3 Platforms Editor

In this section, the user now can start to add elements to the scenario. The first step is to create Platforms. Platform Types need to already be established, and now instances of platform types can be created as nodes in the scenario. Once the platforms (air vehicles, ground vehicles, dismounted persons) are added to the scenario as platforms – then they can be configured with radios.

8.3.1 Create new platform

1. Click on the 'Create new platform' button on the 'Platforms' tab.

2. The 'Platform Configuration' window will pop up.

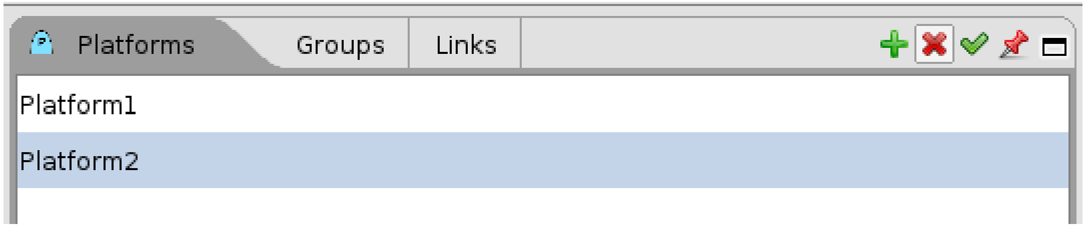


3. Input platform name in the 'Display Name' field.
4. Input starting yaw in degrees in the 'Starting Yaw' field.
5. Input starting pitch in degrees in the 'Starting Pitch' field.
6. Input starting roll in degrees in the 'Starting Roll' field.
7. Set platform type from the 'Platform Type' drop down menu.
8. Add radios to the platform (see Section 8.3.4)
9. Click 'Save' to save the new platform.

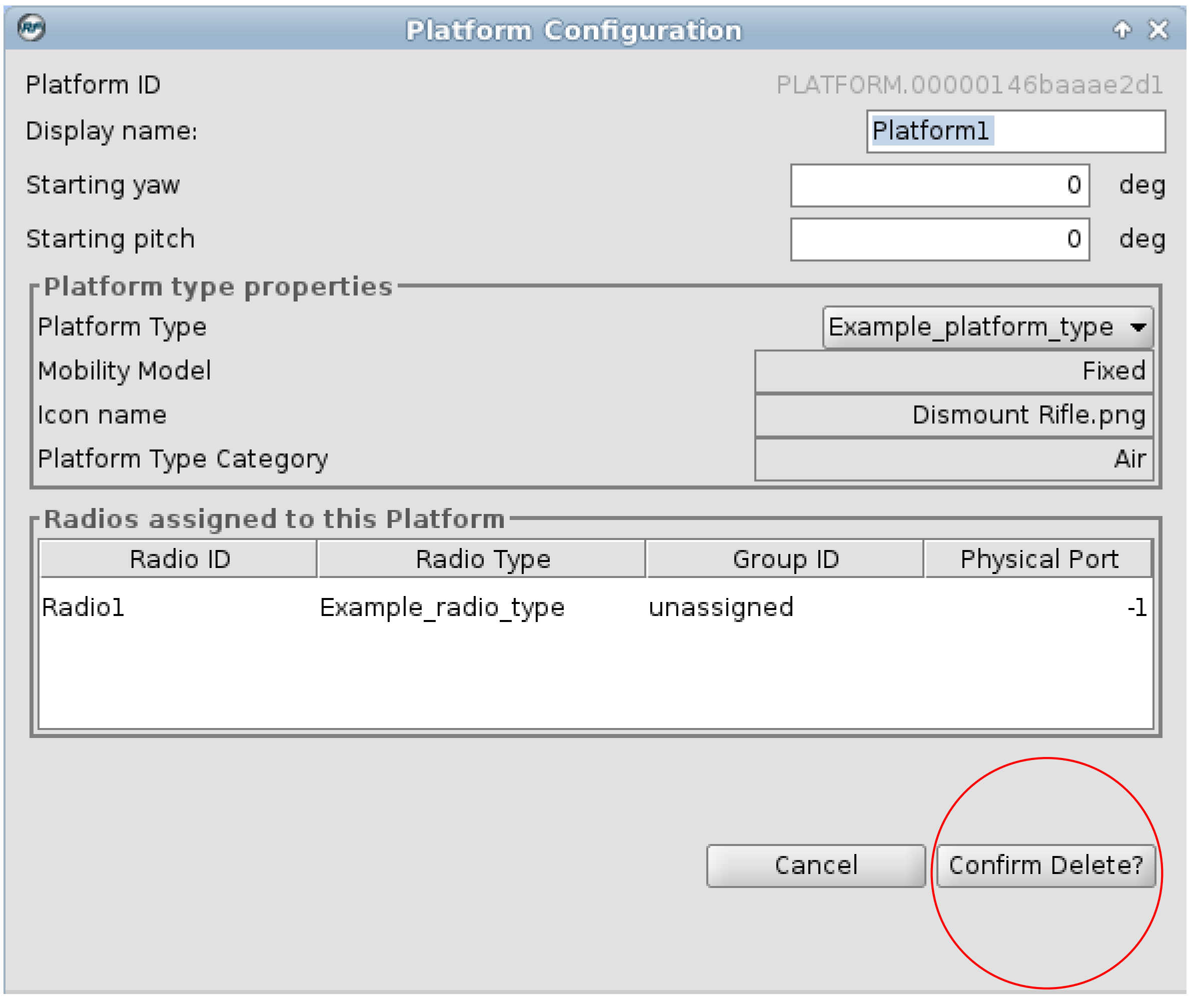
8.3.2 Delete platform

1. Select a platform from the 'Platforms' tab.

2. Click the red ‘x’ icon.



3. Click the ‘Confirm Delete?’ button to confirm removal of the platform

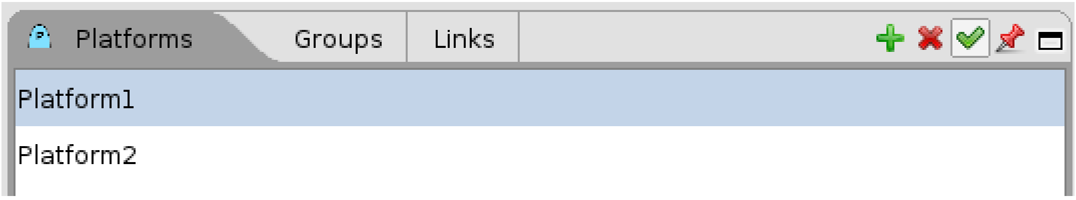


8.3.3 Edit platform

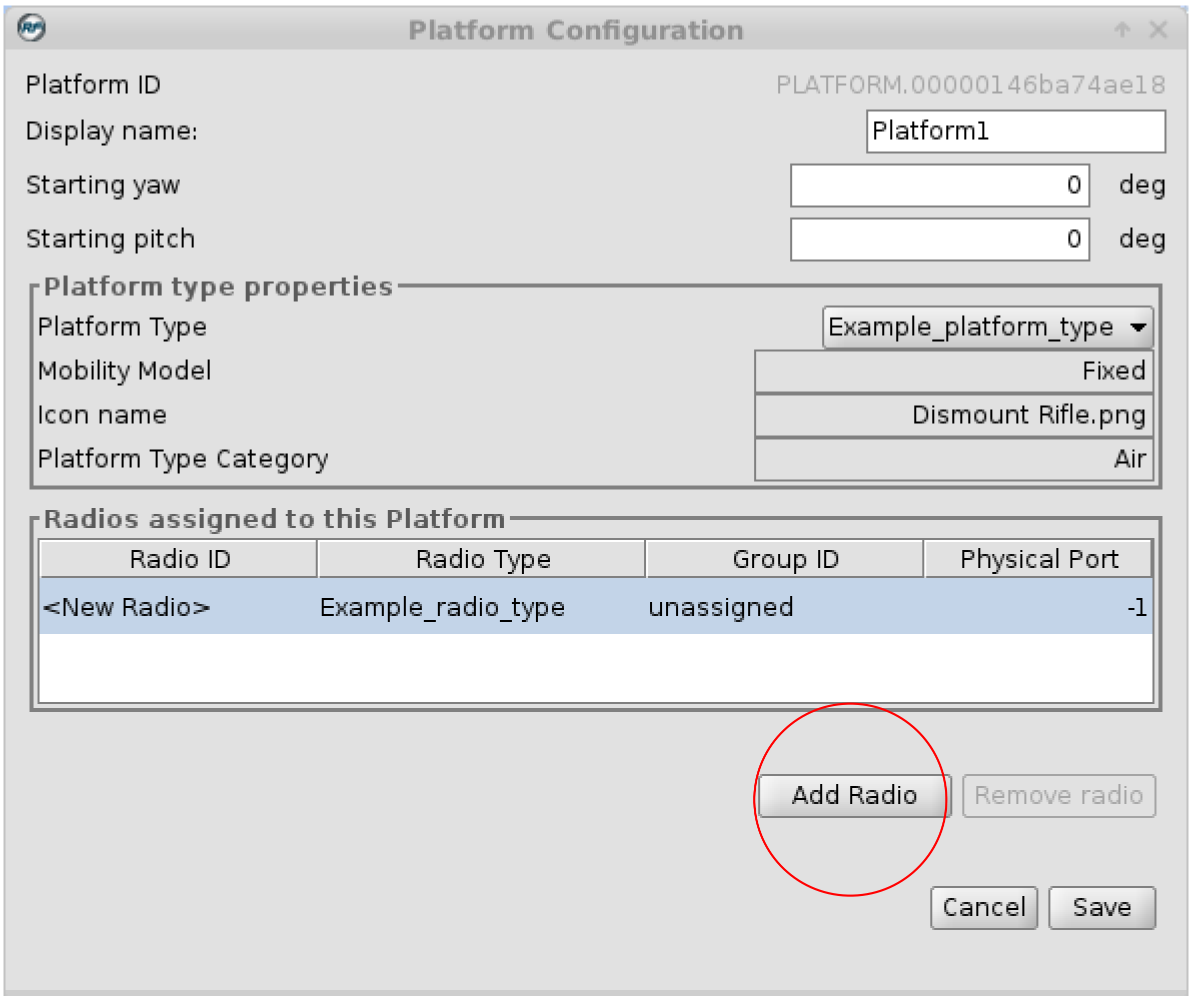
1. Select a platform from the ‘Platforms’ tab.
2. Click the green ‘check’ icon or double click the selected platform.
3. Edit platform.
4. Click ‘Save’ to save changes.

8.3.4 Add radio to platform

1. Select a platform from the 'Platforms' tab.
2. Click the green 'check' icon or double click the selected platform.



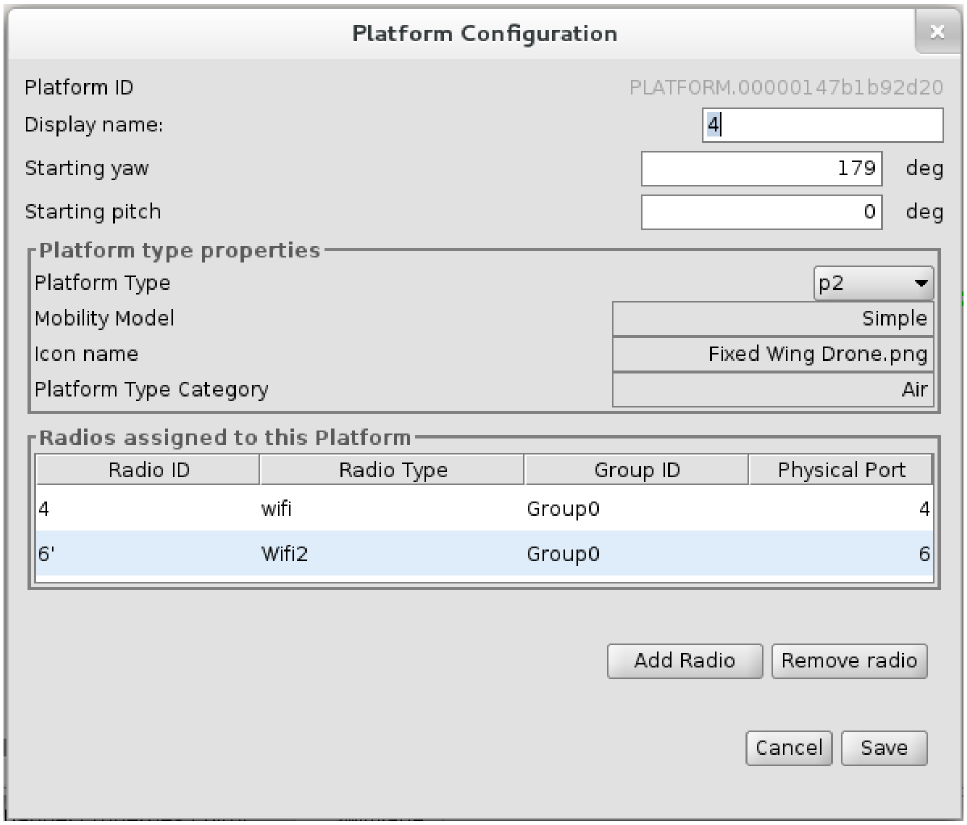
3. Click 'Add Radio' in the 'Platform Configuration' window.



4. A new radio will be added to the 'Radios assigned to this Platform' list.
5. Too add additional radios click 'Add Radio' button again.
6. Double click on the <New Radio> field in the 'Radio ID' column to change the name of the radio.
7. Double click on the field under the 'Radio Type' column to set the radio type from the drop down menu.
8. Click 'Save' to same platform.

8.3.5 Placing Multiple Radios at Different Locations on a Platform

It is often desirable to place radios on a platform at different locations, for example to allow RFnest to accurately represent the relative phases of the signal each radio receives in this case. To accomplish this, multiple radio types must be created, one per unique offset/orientation relative to the platform. For example, if one WiFi radio is attached on each wing of a certain type of remote controlled airplane, two radio types must be created with identical settings except for their lateral offsets. The platform must then have two radios attached, each with the respective types as shown in the Figure below where radio 4 has type “wifi” and radio 6′ has type “wifi2”.

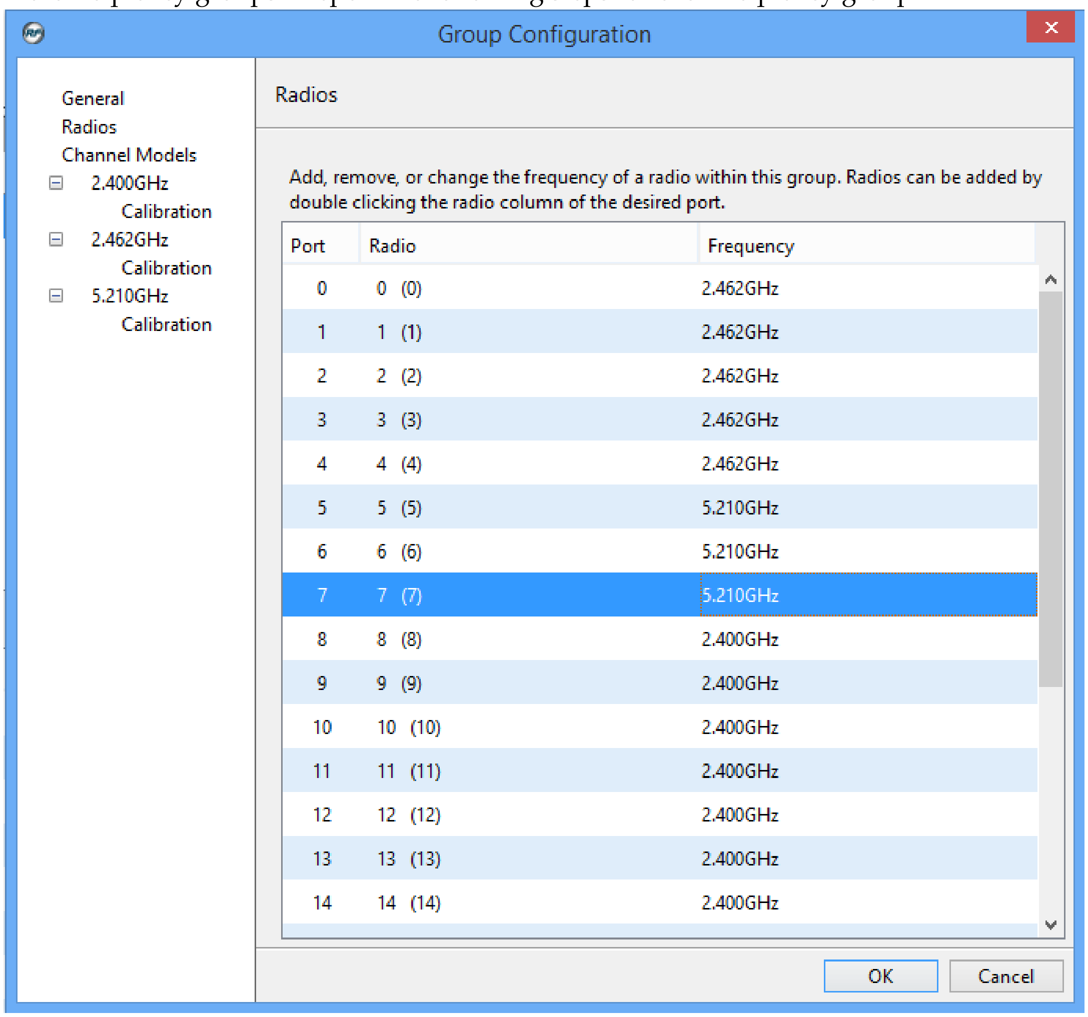


8.4 Group Configuration

The RFnest system needs to know a priori if radios should be able to talk to each other (when the channel models allow). Only radios in the same group will have the channels between them calculated and applied to the transmitted signals. Groups can be further divided into “Frequency Groups,” by assigning radios configured to be sampled at different center frequencies into the same group. The channels between radios in different frequency groups (as well as in different groups) are always set to maximum loss. In addition to assigning radios to groups, the group configuration also allows the user to specify the channel models and the calibration on a per frequency group basis.

1. Click on the ‘Groups’ tab.
2. Double click on a group or select a group and click the green check icon.

3. In the ‘Group Configuration’ window double click on a field in the ‘Radio’ column to assign radio to port. (Only select the radios that should be a part of this particular group). The frequency can be edited in this dialog to use a different RF frequency than that configured for the radio type. Changes in this dialog will automatically populate the tree on the left side under Channel Models with one or more frequency groups. Repeat the following steps for each frequency group.



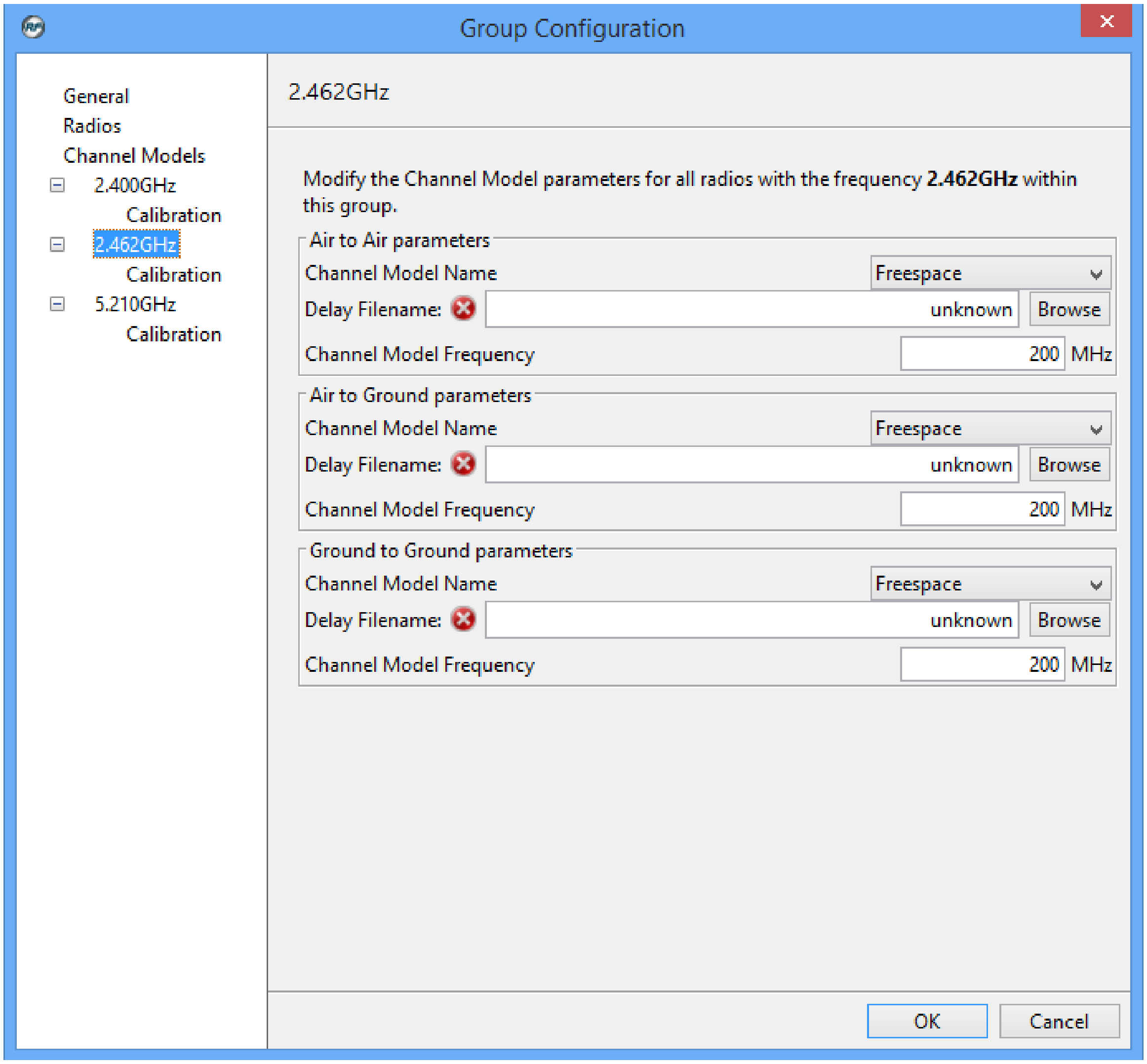
Note: If using the A-Series, radios could be set to any frequency within the specifications. The standard version is 1.8GHz to 2.8GHz, with best performance achieved from 1.8GHz to 2.5GHz. The frequency in this dialog is not applicable to the A-Series.

4. For each link sub-type (air to air, air to ground, ground to ground):

   (a) Set the model from the ‘Channel Model Name’ drop down menu for each type of communications.

   (b) **D-Series Only** Click ‘Browse’ to select and set a delay profile saved on RFview PC in the ‘Delay File Name’ field.

(c) Set the 'Channel Model Frequency' field to the frequency to be used to model the channels. This frequency can be different from the RF frequency, for example if frequency converters are used to provide lower frequency signals into the RFnest hardware, whereas the radios under test would normally transmit over the air via a higher frequency.



5. Click on the calibration sub-item below the frequency group. Configuration is explained in more detail in Section 9.6.

6. Set the 'User Transit Power Offset' to indicate the additional loss or reduced transmit power present at the radios in this frequency group. If this value is set to N, the first N dB of loss will be assumed to have already been applied by the user and subtracted from the channel applied digitally.

7. **(D-Series Only)** Set the Output Attenuation to the desired output analog attenuation. This serves to lower the power level sent to the radio in 10 dB increments, effectively shifting the dynamic range of the system. The specified output attenuation will be automatically adjusted based on per-port system calibration data before being applied to the hardware. Adjusting this value over-writes any manual settings of the output gain field via the Port Status tab.

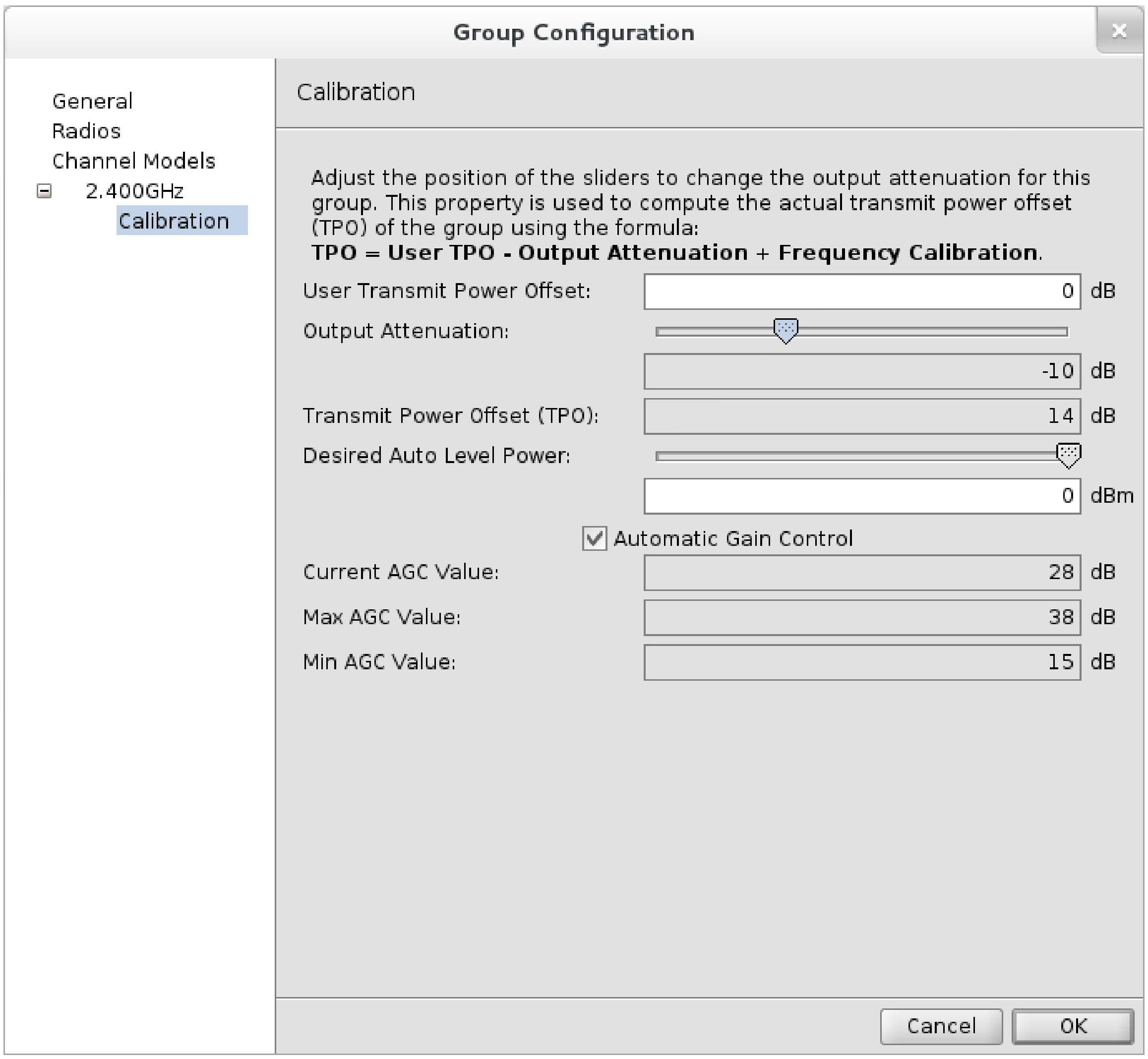
8. **(D-Series Only)** The Frequency Calibration is automatically calculated based on the system calibration data to compensate for the RF frequency response of the system.

9. The Transmit Power Offset (TPO) used in the scenario is then calculated automatically.

10. **(D-Series Only)** The Desired Auto Level Power should be set to the target value used by the Signal Tool's auto-level function and Automatic Gain Control (if enabled). Typically the value should be

left at 0 dBm.

11. **(D-Series Only)** If Automatic Gain Control is desired, check the box to enable it. This function is for radios or other devices where their transmit power varies. See Section 11.4 for instructions **before** enabling this feature, interpreting the AGC values, and resolving errors indicated in this dialog if AGC is enabled.
12. **(D-Series Only)** Current AGC Value - the highest current input gain of any port in the frequency group.
13. **(D-Series Only)** Max AGC Value - the maximum value that the Current AGC value can be increased to while maintaining correct operation.
14. **(D-Series Only)** Min AGC Value - the minimum value that the Current AGC value can be increased to while maintaining correct operation.



15. Click 'OK' to save group configuration. Note that changes to this dialog are not applied in any way until RFview is re-initialized.

8.5 Map controls/options

Map control and options are located on the top right corner of the map.

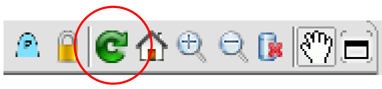
1. To lock the map view and follow a platform select a platform and click on the blue 'Platform' button on the map toolbar.

2. To have the map zoom in and out to show all platforms throughout the scenario click on the 'Lock' button on the map toolbar.

3. To refresh the contents of the map click on the green 'Refresh' button on the map toolbar.

4. To center the map around all existing platform click on the 'Home' button on the map toolbar.

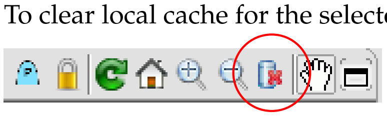
5. To zoom in click on the 'Zoom In' button on the map toolbar or use mouse wheel.

6. To zoom out click on the 'Zoom Out' button on the map toolbar or use mouse wheel.

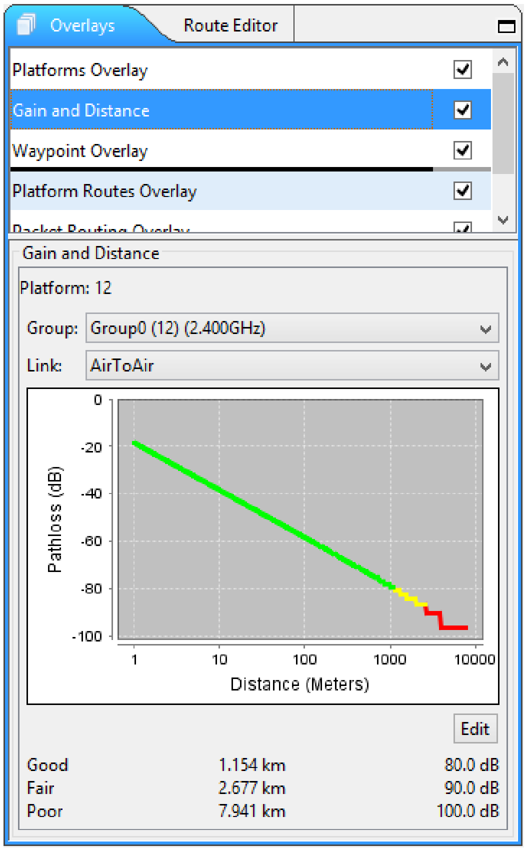
7. To clear local cache for the selected layers click on the 'Clear local cache' button on the map toolbar.

8.6 Overlay Tab

The Overlay tab allows the map overlay layers to be enabled/disabled, reordered, and configured.

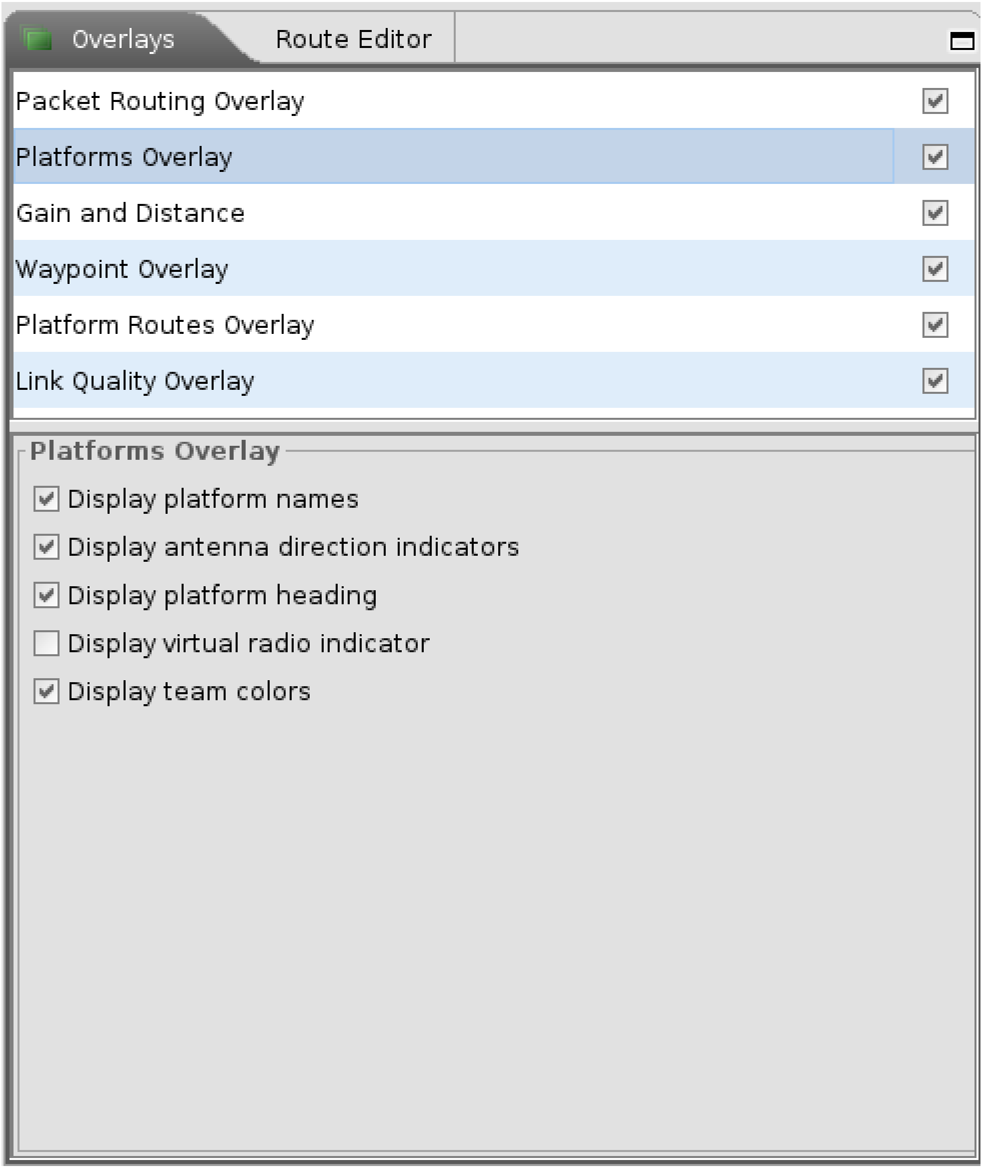
1. To enable/disable an overlay layer, click on the checkbox next to the overlay layer's name in the list. This is useful if a particular overlay is not relevant or creates too much visual clutter.
2. To re-order the overlay drawing order, simply click on and drag the overlay name to a new position in the list. Overlays are drawn in order from top to bottom, and some overlays may partially obscure others. The order configuration provides a means to address conflicts/clutter without completely disabling an overlay layer.
3. To view/edit the settings for an overlay, select the overlay and size the overlay tab such that the settings below the list pane are visible.



8.7 Platforms Overlay

This menu allows the user to configure how or if certain data is displayed on RFview. The user can use this menu to select and deselect platform information for display while the scenario is running.

1. Select 'Platforms Overlay' from the 'Overlays' tab.



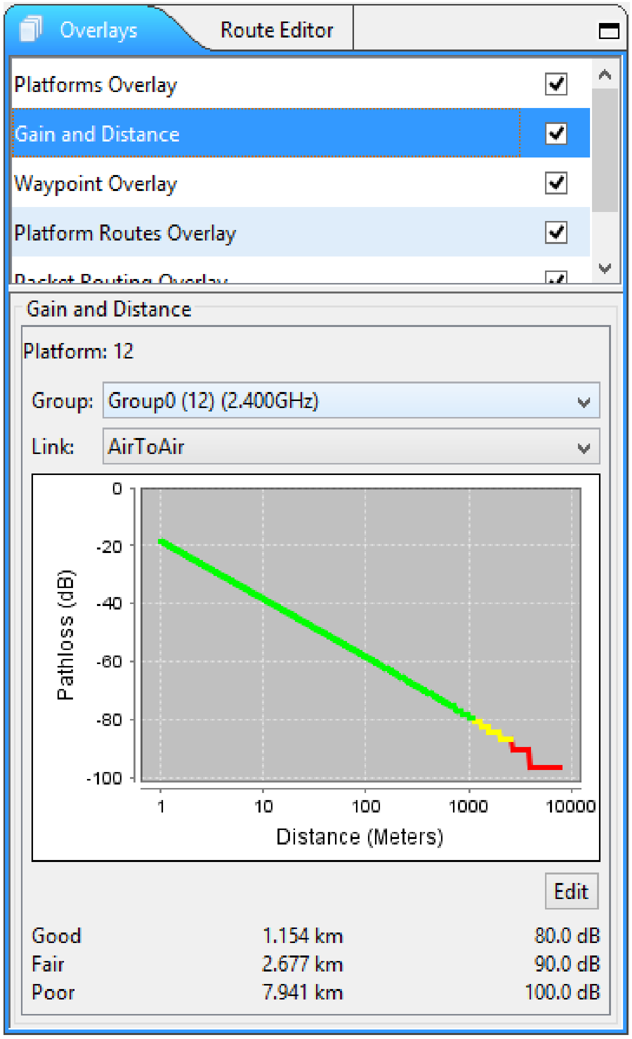
2. Checking or unchecking 'Display platform names' checkbox will show or hide platform names on the map.
3. Checking or unchecking 'Display antenna direction indicators' will show or hide radio antenna direction arrows on the map.
4. Checking or unchecking 'Display platform heading' will show or hide platform orientation on the map.
5. Checking or unchecking 'Display virtual radio indicator' will show or hide the virtual radio icon on the map.
6. Checking or unchecking 'Display team colors' will show or hide the fill color around the platforms. Note that the team color for each platform is configured in the Route Editor tab.

8.8 Gain vs Distance Overlay

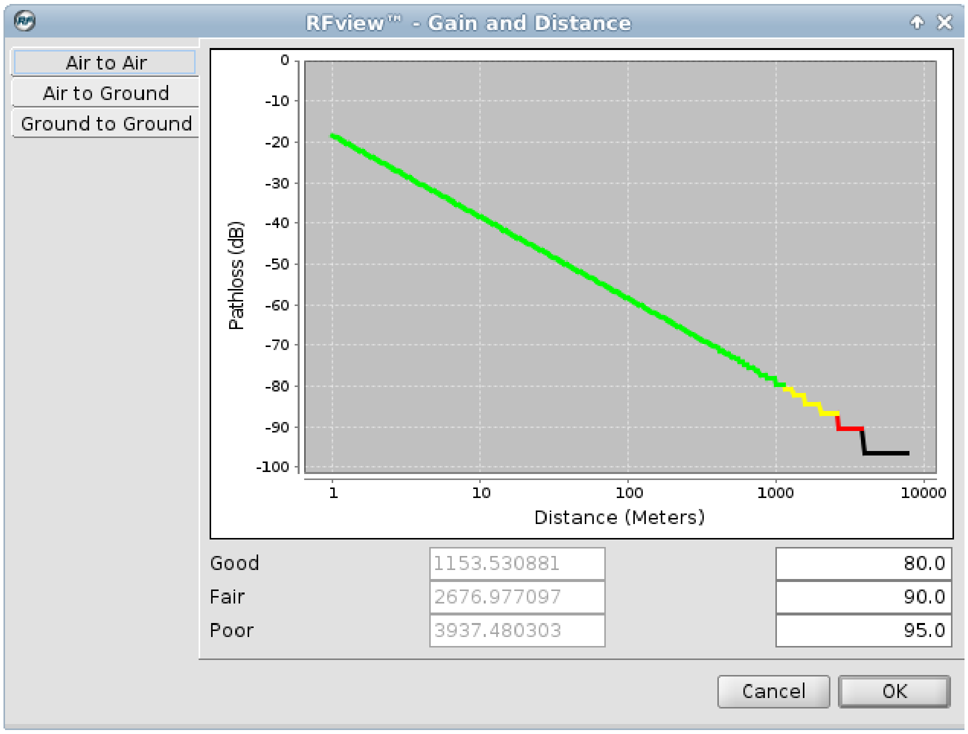
The Gain vs. Distance Overlay is typically used when setting up a scenario to see if radios may or may not be able to communicate with each other. This menu allows the user to turn on / off the Gain vs. Distance

overlay on RFview. In addition, this menu enables the user to configure the various signal strength rings based on the radio characteristics. The Gain vs. Distance overlay configuration is done on a per frequency group and per link type basis.

1. Select 'Gain vs Distance' from the 'Overlays' tab.
2. Check or uncheck the 'Gain vs Distance' checkbox to show or hide the signal range indicator.



3. Select a platform in the Map tab or in the Platform tab. If no platform is selected, the configuration will not be enabled.
4. Select the Frequency Group from the 'Group' drop-down menu. Only valid frequency groups for the selected platform are shown (i.e., those which its radios are members of).
5. Select the link type from the 'Link' drop-down menu. Only valid link types for the selected platform are shown (e.g. an air platform will not have ground to ground as an option, air to air will not be an option if no other air platforms exist in the frequency group, etc.).
6. View the graph of gain vs distance and the circular thresholds in the map. Note that the channel model does not necessarily create perfectly circular gain patterns. These circles ignore any terrain effects, platform orientations, and assume optimal antenna patterns.
7. Use/edit the guides to help layout the scenario as needed. See Section 9.8 for more details. Click 'Edit' to set the signal strength thresholds for the currently selected Frequency Group.

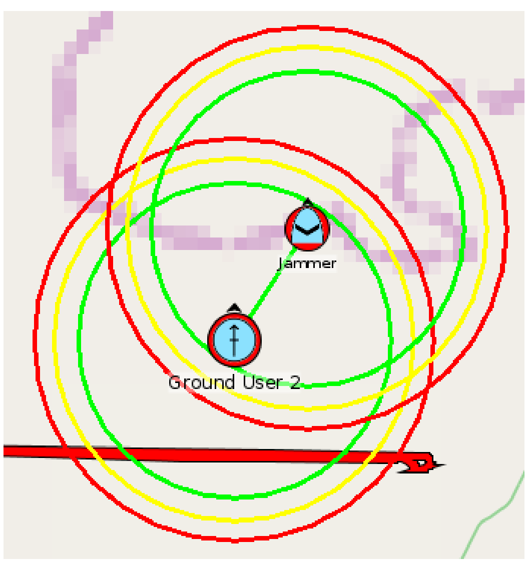


8. Set the 'Good', 'Fair', and 'Poor' threshold of Pathloss (dB) to desired values based on user observations/knowledge. Note that these thresholds do **NOT** control radio behavior in **ANY WAY**. These thresholds serve **ONLY** as an estimated visual reference to assist the user in planning and understanding the scenario.

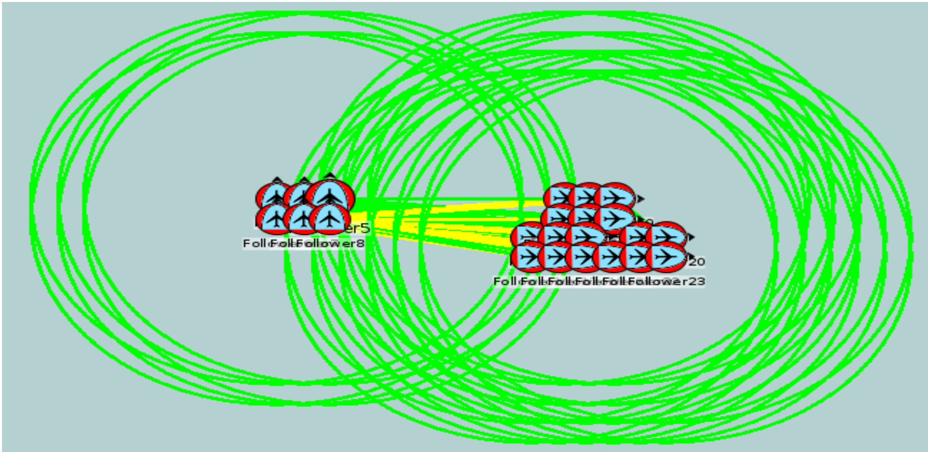
9. Click 'OK' to save settings or 'Cancel' to cancel.

   The setting will reflect the display of signal strength indicators on the map, showing where to position the nodes on the map in order to get desired channel values.

   In the screenshot below, two platforms are within each other's "good" (green) range, indicated by their being within each other's green circle and the link being colored green.



   In the following screenshot, the thresholds differ more significantly (both in dB and in meters), some platforms are within each other's good/green range, while others are just outside.



NOTE #1: The Gain vs. Distance circular overlay and the link colors are intended as a guide and visual reference ONLY. Configuring these thresholds correctly (or choosing not to configure at all) is purely for the benefit of the user and has no bearing on the channel properties or the performance of the system under test. Nodes being within green circles or having a green link does not guarantee in any way the link is actually good. These tools are simply a more visual way to represent the user's knowledge that, for example, a link with a gain above -100 creates a "good" link. If the thresholds are not configured, it is recommended to disable these overlays to reduce clutter.

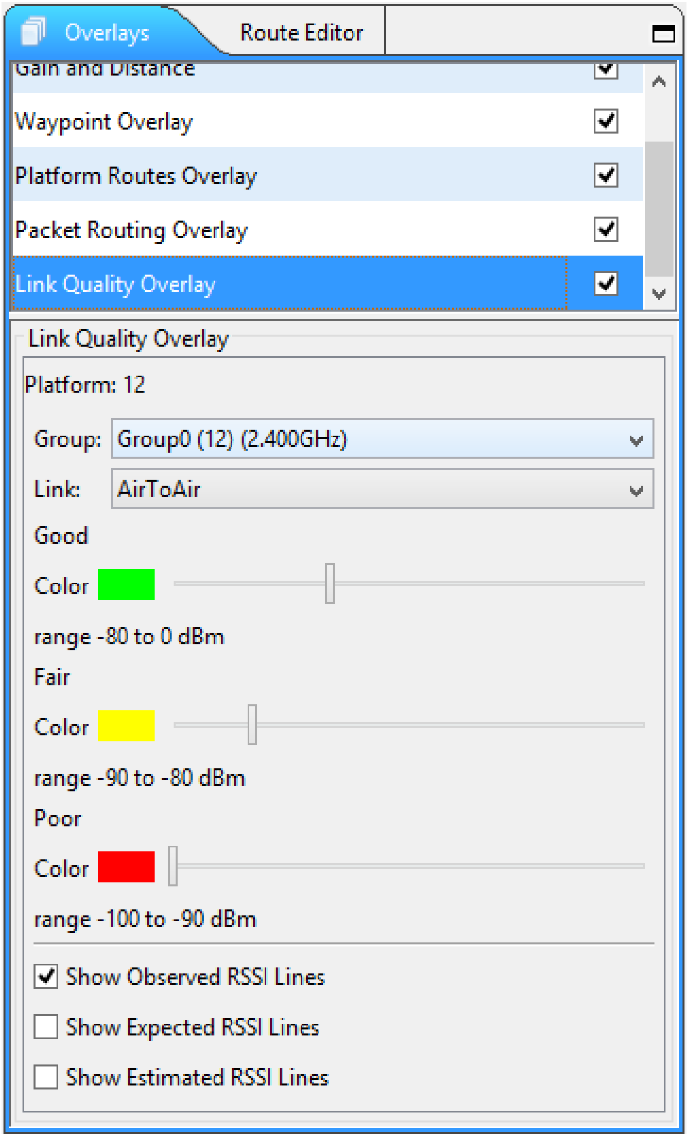
NOTE #2: The Gain vs Distance circles are a very rough approximation (ignoring terrain and antenna). The link coloring, however, is based on the actual gain being applied to a link and is much more accurate. In some cases, especially when a platform is near the boundary of a Gain vs. Distance circle, the link color may not correspond with the circle a platform is within. In all cases, trust the link color (and the gain shown in the Channel Properties Editor Tab) over the Gain vs. Distance circles.

NOTE #3: Only the Gain vs. Distance circles for the selected platform's frequency group and nodes it communicates directly with via the selected link type are shown. However, all links are shown (when enabled).

8.9 Link Quality Overlay

This menu allows the user to configure the link quality lines (setting the color and thresholds), and also allows the user to configure if the link quality lines should be on or off. The settings are done per radio.

1. Select the 'Link Quality Overlay' field from the 'Overlays' tab.
2. Check or uncheck the 'Link Quality Overlay' checkbox to show or hide link quality indicators on the map.



3. Set the Group from the 'Group' drop down menu.
4. Set the link to 'AirtoAir', 'GroundToGround', or 'GroundToAir' from the 'Link' drop-down menu.
5. Select the color for the 'Good', 'Fair', and 'Poor' link quality indicator by using the sliders. Only the colors can be edited in this dialog. The thresholds for determining when to color a link are configured in the Gain vs Distance Overlay.
6. Check or uncheck the desired checkboxes to show or hide the 'Observed', 'Expected', and 'Estimated' RSSI. If unsure, use the Expected option. These options are described below

   Observed Link Quality:

   - This value must be reported by the radios or other portions of the system under test.
   - To show RSSI in RFview, the user can send periodic updates to RFview's multicast address: 224.1.2.208 on port 20860 using the format specified in the API.
   - The RSSI indicator will be displayed until a new update is sent.

   Expected Link Quality:

   - Enable this link quality if unsure, nothing else is available, radios are not (yet) connected, or the hardware is not present.

- The expected link quality is the quality that is expected to be present on the link based on the configured transmit power offset for the corresponding frequency group and the channel gain that was calculated by the CEC, specified manually by the user via the Channel Properties Editor, or fed into the system via a DCU Request or CCR message.

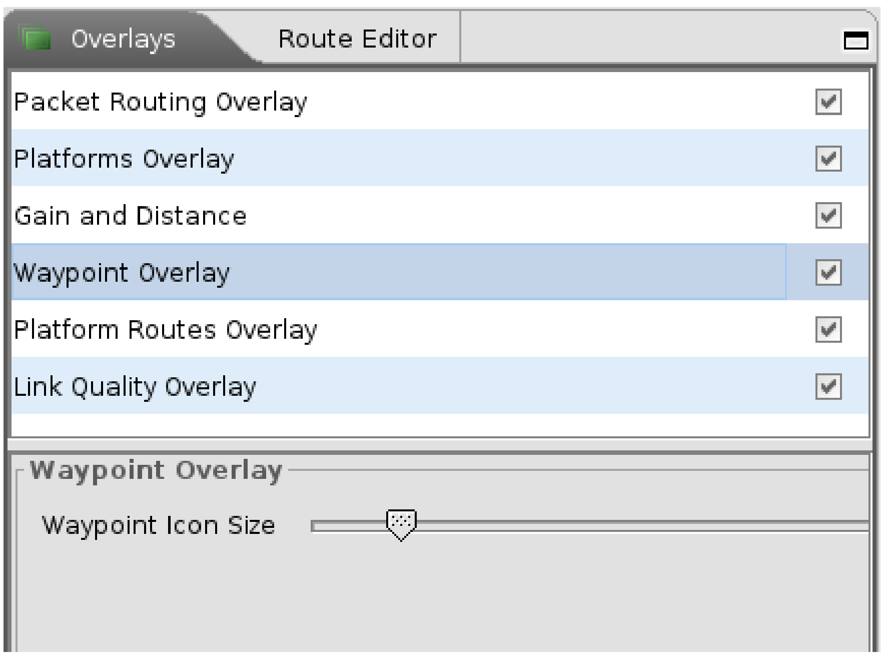
Estimated Link Quality:

- The estimated link quality improves upon the correctness of the expected link quality by also considering the actual power at which the Rx port for each link's radios are transmitting.

8.10 Waypoint Overlay

This menu allows the user to configure the size of waypoints for when the waypoints are displayed on RFview.

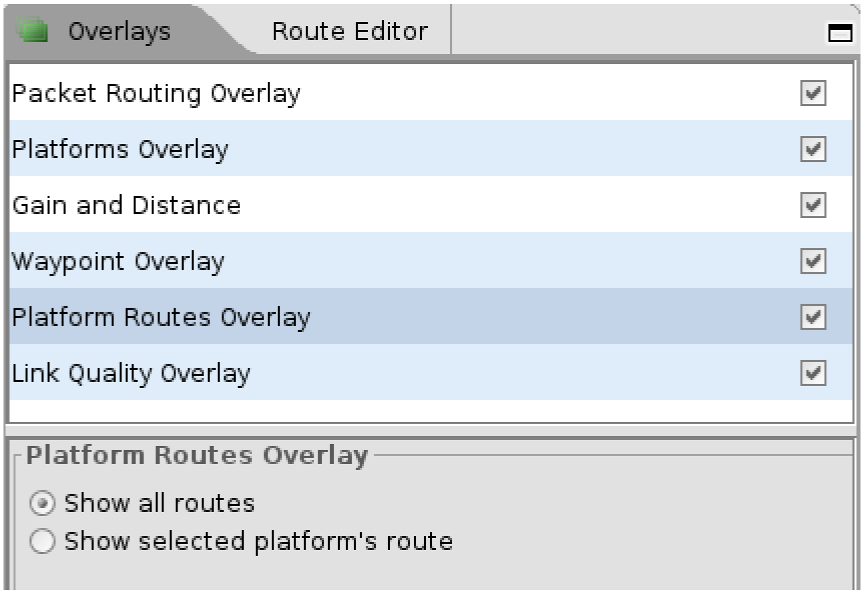
1. Select the 'Waypoint Overlay' field from the 'Overlays' tab.
2. Check or uncheck the 'Waypoint Overlay' checkbox to show or hide waypoints on the map.
3. Drag the 'Waypoint Icon Size' slider to change the size of the waypoints that appear on the map.



8.11 Platform Routes Overlay

This menu allows the user to select which platform routes (if any) should be displayed on RFview. Typically routes will be displayed during initial scenario set up and editing, but hidden afterward.

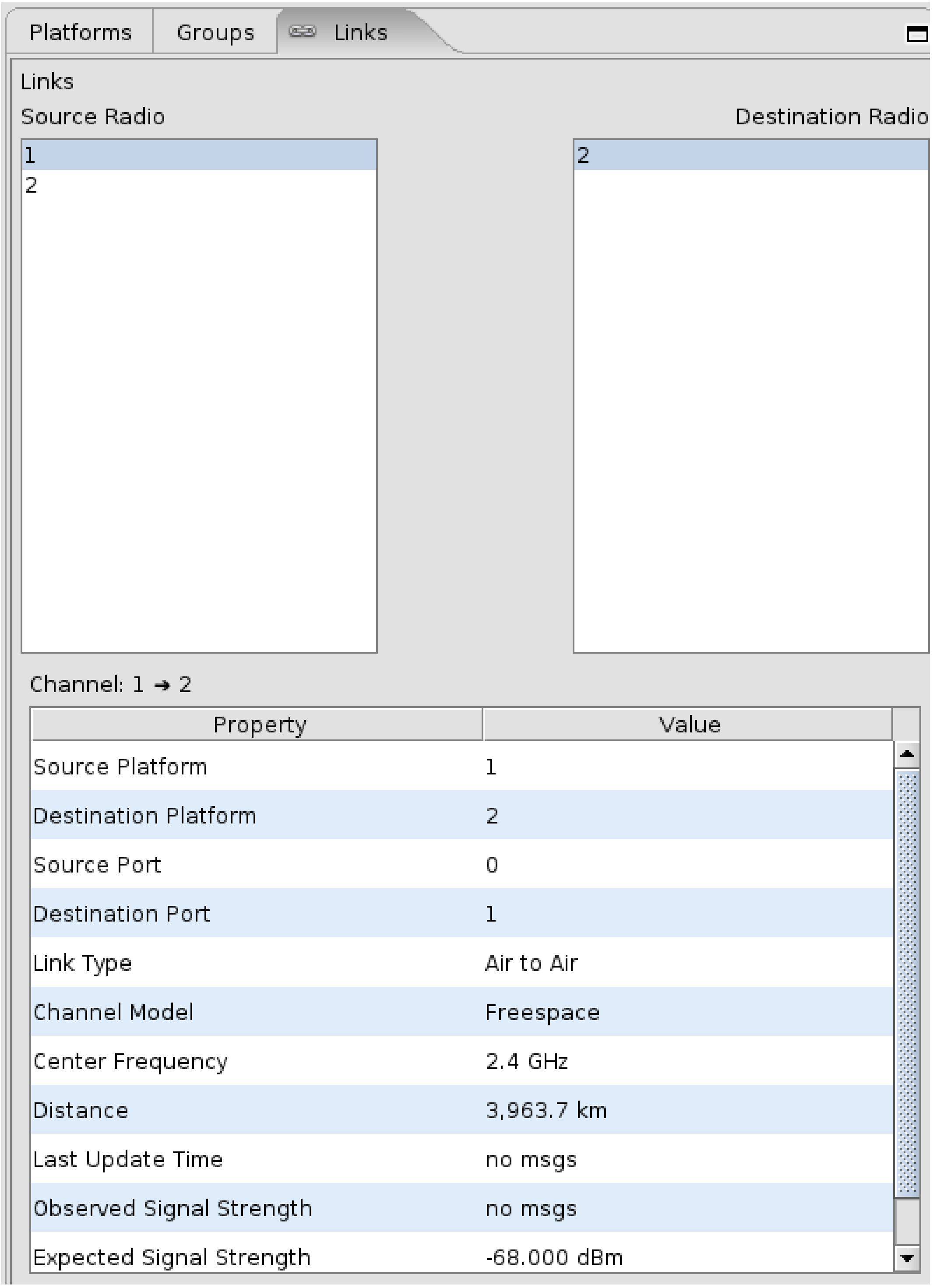
1. Select the 'Platform Routes Overlay' field from the 'Overlays' tab.
2. Check or uncheck the 'Platform Routes Overlay' checkbox to show or hide routes on the map.
3. Select one of the radio buttons to either show all routes or show only selected platform's route.



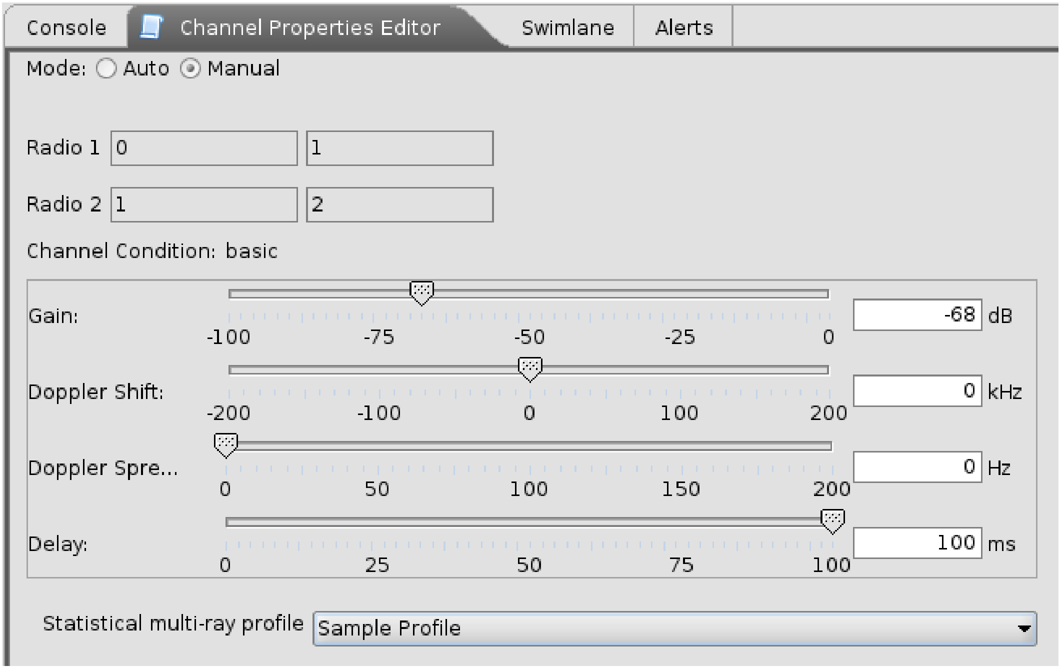
8.12 Links/Channel Properties Editor

The channel properties editor enables the user to set custom values for channel properties that would overwrite any properties set by the model.

1. Select the 'Link' tab.



2. Select two radios whose channel is to be set to custom values.
3. Select the 'Channel Properties Editor' tab.



4. Set the mode radio button to 'Manual'.
   - In 'Auto' mode the tab will show the channel conditions calculated by the software model.
   - In 'Manual' mode the user can set custom values for channel properties and the model calculations will be ignored.
5. Set desired gain by moving the 'Gain' slider or typing a number in the provided field.
6. Set desired doppler shift by moving the 'Doppler Shift' slider or typing a number in the provided field. Note: this field is only present if supported by the current RFnest hardware.
7. Set desired doppler spread by moving the 'Doppler Spread' slider or typing a number in the provided field. Note: this field is only present if supported by the current RFnest hardware.
8. Set desired delay by moving the 'Delay' slider or typing a number in the provided field. Note: this field is only present if supported by the current RFnest hardware.
9. Set statistical multi-ray profile by selecting one from the 'Statistical multi-ray profile' drop down menu. Note: this field is only present if supported by the current RFnest hardware.

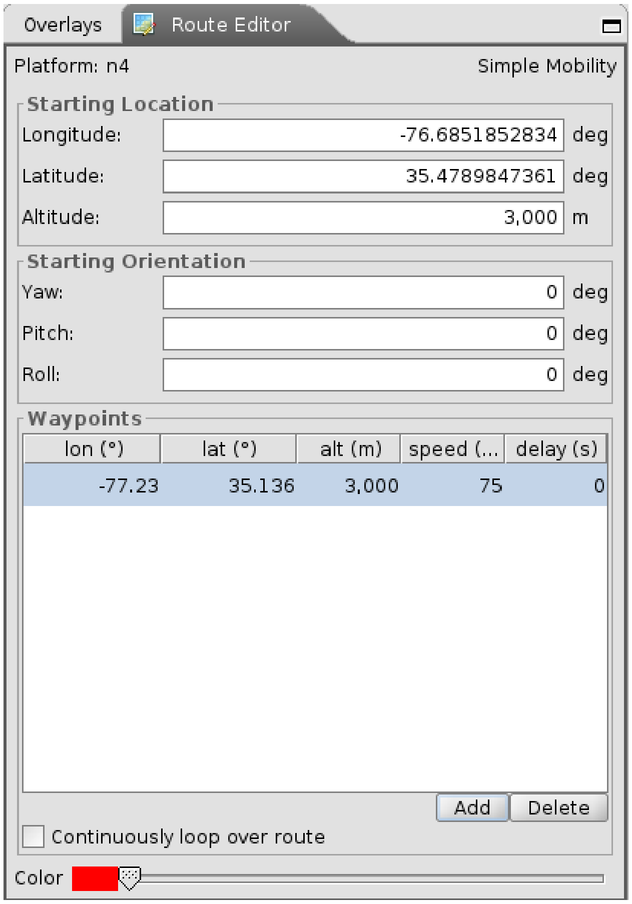
NOTE: Be sure to change any links back to manual if they are no longer being manually controlled/over-ridden. Otherwise, the mobility of platforms will NOT cause the link's parameters to be updated as nodes move in the scenario.

8.13 Route Editor

The Route Editor enables the user to manually enter the node waypoints over time for the scenario. The starting position is specified along with the starting velocity and orientation. Multiple waypoints and the final position can be added as well. Note that the platform will stop suddenly when the end of the route is reached, and so the user should be sure to specify sufficient waypoints for the entire scenario duration.

1. Select a node from the 'Platforms' tab.

2. Click on the 'Route Editor' tab.

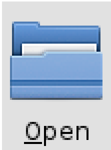


3. Drag the node by clicking and holding the left mouse button to set the starting node position on the map, or set the starting location and orientation in the 'Route Editor' tab by editing the 'Longitude', 'Latitude', 'Altitude', 'Yaw', and 'Pitch' fields.
4. Click 'Add' to add a waypoint. (On the map, the selected Way Point will be temporarily enlarged, and the user can then drag and drop it).
5. Left click and hold to drag the waypoint on the map to position it, or modify the 'lon', 'lat', 'alt', 'speed' (m/s), and 'delay' (seconds to wait at a waypoint before continuing route) columns in the 'Waypoints' editor window.
6. To have the platform continuously loop over the set waypoints set the 'Continuously loop over route' option. (By default, the platform will go from the start to the finish point and then stop.)
7. To change waypoint and route color use the slider at the bottom of the 'Route Editor' tab.
8. To delete a waypont, select a row from the 'Waypoints' editor and click 'Delete'.

8.14 Saving/Loading scenarios

1. To save a scenario go to the 'Scenario' menu and select 'Save' or click on the 'Save' icon on the toolbar. Browse to the destination folder where you want to save the scenario.

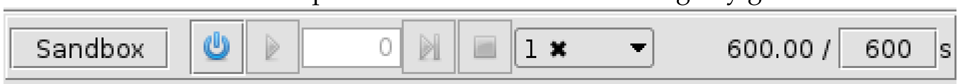
2. To load a scenario go to the 'Scenario' menu and select 'Open'. Browse to the destination where the .xml scenario file is saved and load the scenario.



8.15 Scenario Record/Replay

Typically, a user will generate or play a "canned" scenario by using this scenario Record/Replay button. Generating a "canned" scenario captures all of the channel models and events for later analysis, visualization, and/or demonstration. This section details how the user records and then replays these scenarios.

1. Click on the 'Sandbox' button and select 'Record'. When the Sanbox/Record/Playback option is set to 'Record' RFview will capture all data and events during any given scenario.



2. Initialize the scenario and run.

3. The scenario recording will stop when the scenario has ended, otherwise user can stop the recording and the scenario when needed by pressing the 'stop' button.
4. The recording is automatically saved. If a recording was already saved for the current scenario, the previous recording will not be over-written.
5. Click on the 'Sandbox' button and select 'Playback'.

6. If more than one recording was saved for this scenario, select the desired recording to play.
7. Click the 'Play' button to replay the recorded scenario, data, and all events.

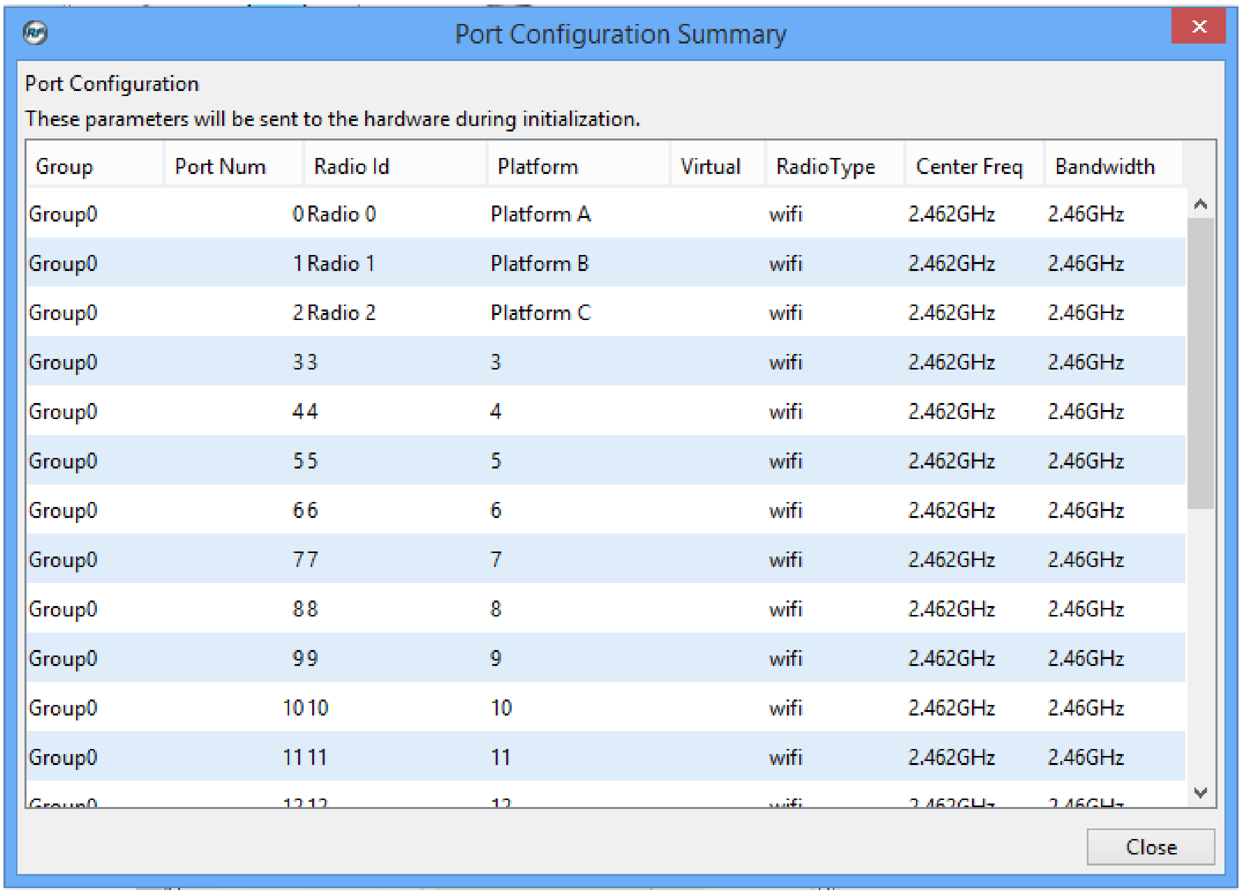
9. Attaching, Verifying, Calibrating, and Operating Radios in the System

Successfully attaching radios to the system requires associating physical hardware ports with radios in the scenario, cabling the radios to the system, setting the transmit power of the radios, verifying input power levels, verifying connectivity, verifying isolation, setting the calibration, monitoring power levels, tuning gain thresholds, and creating alerts to monitor radio behavior. RFview contains a number of features to facilitate this process, however this step requires patience and careful planning. NOTE: Radio transmit powers may fluctuate over time, for example due to overheating, unpredictable behavior, or other radio quirks. The same is true for a radio's receive chain. If the radio appears to be suddenly not transmitting or receiving as expected and the steps in this section do not resolve the problem (or produce results which do not seem to make sense), the radio itself should be verified by connecting it to a spectrum analyzer to verify transmit power or to another radio via an RF cable and appropriate attenuators to verify the radio is sending and receiving correctly.

9.1 Port Configuration Summary

The Port Configuration Summary dialog can be viewed by clicking on the “Ports” icon or going to Hardware, Ports in the menu. This dialog provides a quick reference to the port and frequency each radio in the scenario has been configured to use in the scenario. In order for the scenario to work correctly, the radios must actually be connected to the matching port in this list and the radio must be operating at the frequency configured. While viewing this dialog, double-check that this is the case.

NOTE: If a given radio is not behaving as expected, return to the Port Configuration Summary and re-verify the radio is on the right port and is at the right frequency, as this is a common source of errors.

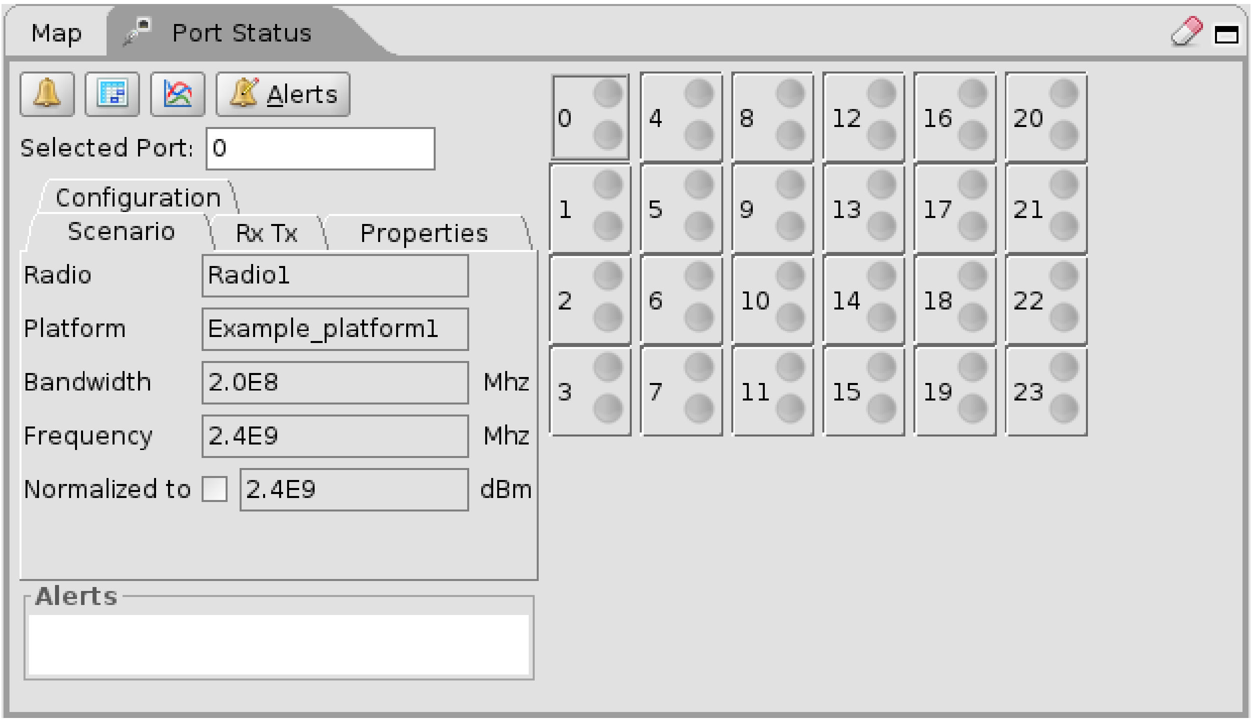


Group	Port Num	Radio Id	Platform	Virtual	RadioType	Center Freq	Bandwidth
Group0	0	Radio 0	Platform A		wifi	2.462GHz	2.46GHz
Group0	1	Radio 1	Platform B		wifi	2.462GHz	2.46GHz
Group0	2	Radio 2	Platform C		wifi	2.462GHz	2.46GHz
Group0	3	3	3		wifi	2.462GHz	2.46GHz
Group0	4	4	4		wifi	2.462GHz	2.46GHz
Group0	5	5	5		wifi	2.462GHz	2.46GHz
Group0	6	6	6		wifi	2.462GHz	2.46GHz
Group0	7	7	7		wifi	2.462GHz	2.46GHz
Group0	8	8	8		wifi	2.462GHz	2.46GHz
Group0	9	9	9		wifi	2.462GHz	2.46GHz
Group0	10	10	10		wifi	2.462GHz	2.46GHz
Group0	11	11	11		wifi	2.462GHz	2.46GHz

9.2 Port Status (D-Series Only)

The Port Status panel helps the user configure and view port-specific parameters as well as to monitor and troubleshoot the system by identifying which ports are currently active (and are seeing RF signals). The user also has the ability to set alerts for specific conditions.

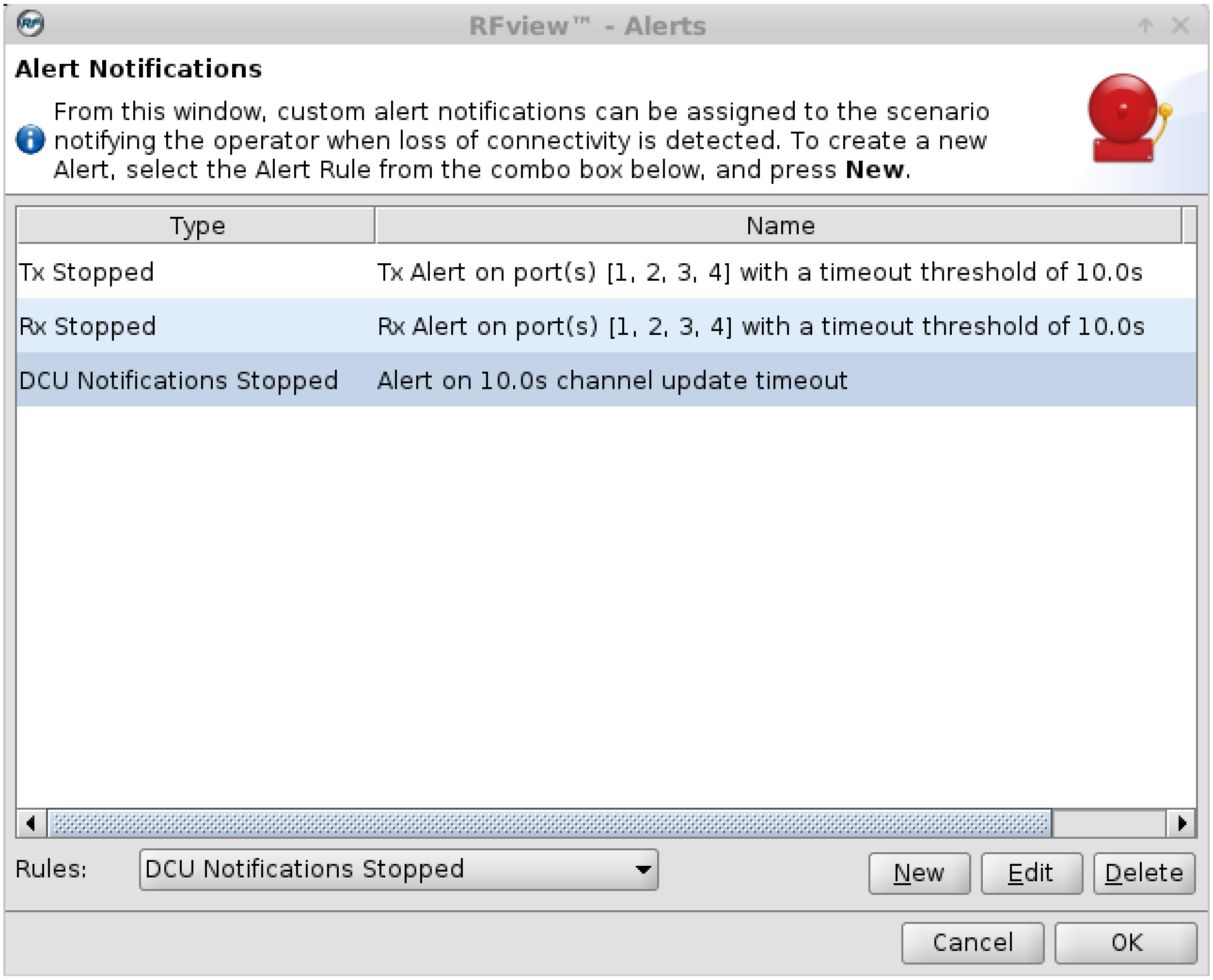
1. Select the ‘Port Status’ tab.



2. On the right half, the activity status for all ports is shown at a high level.
3. Click on the desired port to see the port's status.
4. Click on the 'Scenario' tab to see scenario parameters and settings: which radio is assigned to this port, the platform that this radio/port I assigned to, and the bandwidth and frequency settings.
5. Click on the 'Rx Tx' tab to see the time of time since the last signal was sent and received, and the power and frequency of the last transmitted signal.
6. gains, and configure the long delay (if applicable to that port).

9.3 Alerts

To set alerts, click on 'Alerts' button or use 'Alt + A' keyboard shortcut.

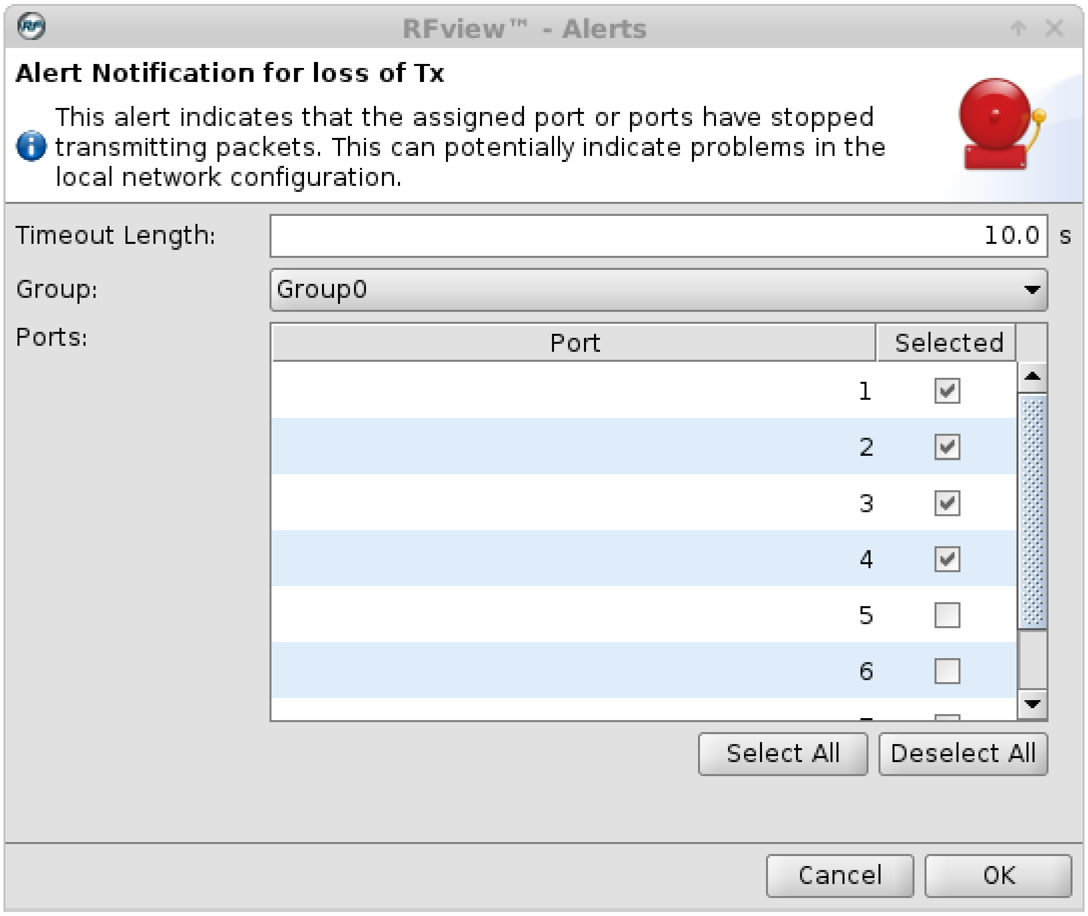


1. Click on the 'Rules' drop down menu to select a type of alert to set.
2. Click 'New' to create a new alert, 'Edit' to modify an existing alert, and 'Delete' to remove an existing alert.

9.3.1 Create 'Tx Stopped' alert (D-Series Only)

This is an alert that will trigger if a specific port will stop sensing a transmit signal.

1. Select 'Tx Stopped' from the 'Rules' drop down menu.
2. Click 'New' to create a new alert.

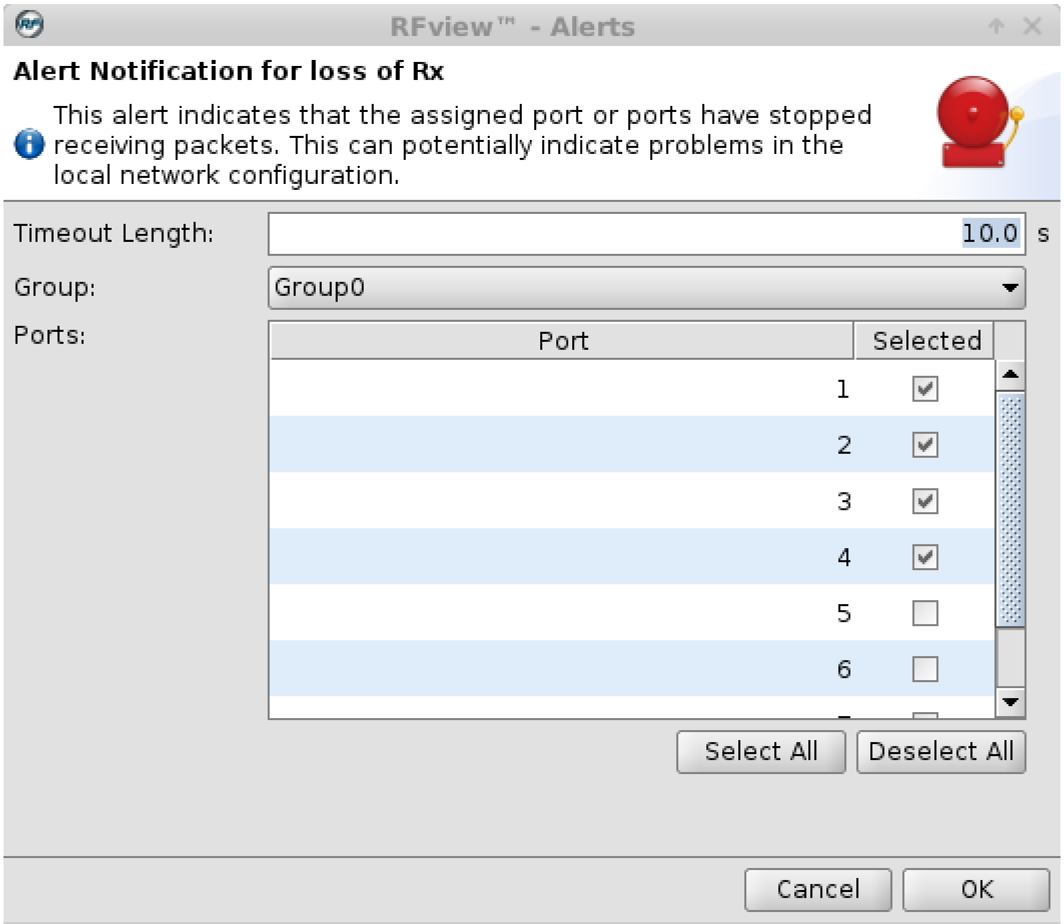


3. Set the length of time without a Tx signal that will trigger the alert in the 'Timeout Length' field.
4. Select the group of ports from the 'Group' drop down menu.
5. Select desired ports by checking the checkboxes in the 'Selected' column of the 'Ports' table.
6. Click 'OK' to save the alert or 'Cancel' to cancel.

9.3.2 Create 'Rx Stopped' alert (D-Series Only)

This is an alert that will trigger if a specific port will stops receiving a signal.

1. Select 'Rx Stopped' from the 'Rules' drop down menu.
2. Click 'New' to create new alert.

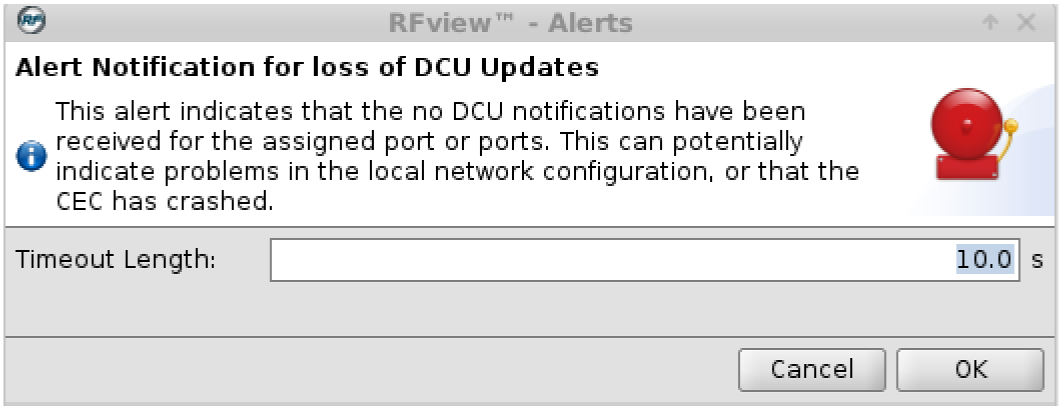


3. Set the length of time without an Rx signal that will trigger the alert in the 'Timeout Length' field.
4. Select the group of ports from the 'Group' drop down menu.
5. Select desired ports by checking the checkboxes in the 'Selected' column of the 'Ports' table.
6. Click 'OK' to save the alert or 'Cancel' to cancel.

9.3.3 Create 'DCU Notification Stopped' alert

This is an alert that will trigger if RFview stops receiving DCU notifications from the CEC. This can potentially indicate problems in the local network configuration, or that the CEC has stopped or crashed.

1. Select 'DCU Notification Stopped' from the 'Rules' drop down menu.
2. Click 'New' to create new alert.

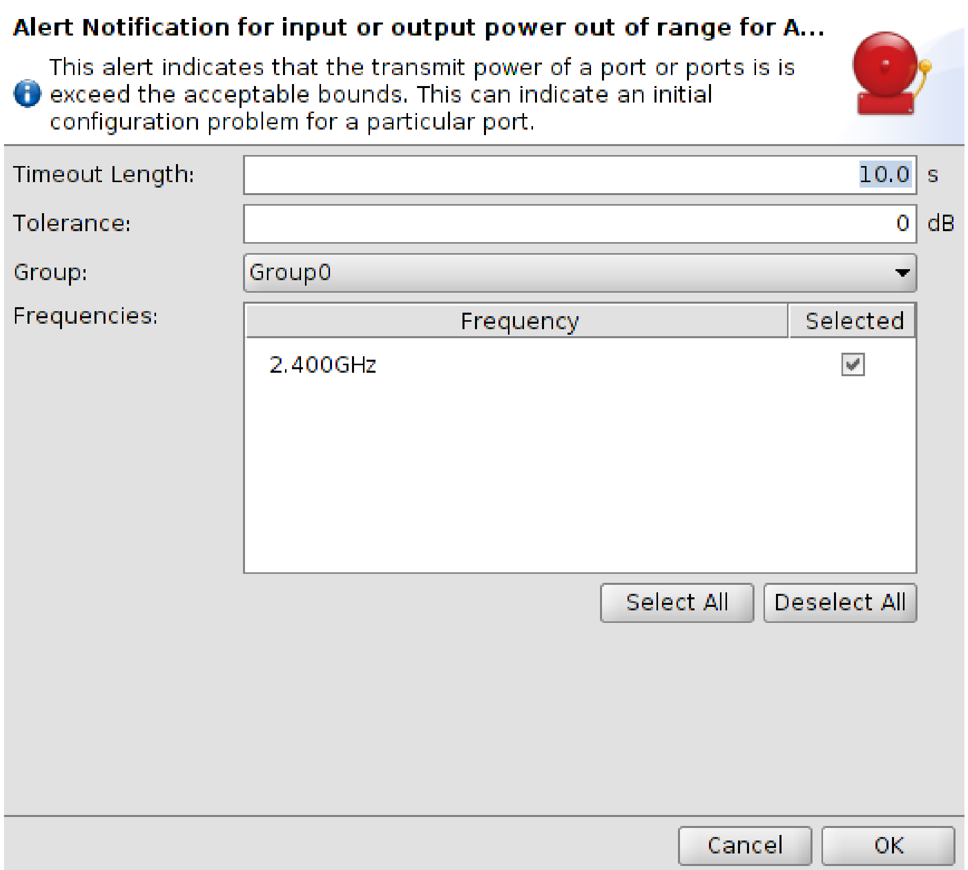


3. Set desired timeout length for the DCU notifications in the 'Timeout Length' field.
4. Click 'OK' to save the alert or 'Cancel' to cancel.

9.3.4 Create 'Automatic Gain Control Out of Range' alert

This is an alert that will trigger if RFview's AGC logic, if enabled, approaches the minimum or maximum value. These values are automatically determined (see Section 11.4). If the AGC approaches the minimum value, an alert is generated for the offending port(s). If the AGC approaches the maximum value, an alert is generated for all ports in the frequency group.

1. Select 'Automatic Gain Control Out of Range' from the 'Rules' drop down menu.
2. Click 'New' to create new alert.



3. Set desired timeout length in the 'Timeout Length' field. The timeout for this alert is the amount of time to wait after the previous alert until potentially generating a new alert for the same frequency group.

4. Set the desired tolerance. The default is 0 dB, indicating the alert will fire only if the minimum or maximum values are reached (or exceeded). If for example a value of 2 dB is entered, the alert will be generated if the AGC is within 2 dB of the minimum or maximum.

5. Select the Group for the alerts. Typically this can be left as Group0.

6. Select the Frequency Group(s) for the alert. Note that if AGC is not enabled for a frequency group, the alert will not be triggered.

7. Click 'OK' to save the alert or 'Cancel' to cancel.

9.4 Recording and Replaying Signals Using the Signal Tool (D-Series Only)

This menu allows the user to capture or playback RF snippets, or display RF signal in real time. Signals can be recorded and replayed in two directions. The "port" direction records signals transmitted by the radio into the Rx port of the hardware, and replays signals into the scenario as if transmitted into that Rx port. The "noise" direction records signals about to be received via the radio attached to the Tx port of that port and replays signals on the outgoing or "noise" path to inject signals directly to the radio via the Tx port.

Note: A signal replayed in the noise direction bypasses the configured channels, only the noise gain parameter for each port affects such signals. A signal being replayed in the noise direction exits the Tx port with the configured noise gains for each port on the relevant DDB. A signal replayed in the port direction passes through the channels, potentially being received by all ports in the frequency group.

To use the Signal Tool:

1. Right click on 'Group 0' from the 'Groups' tab.

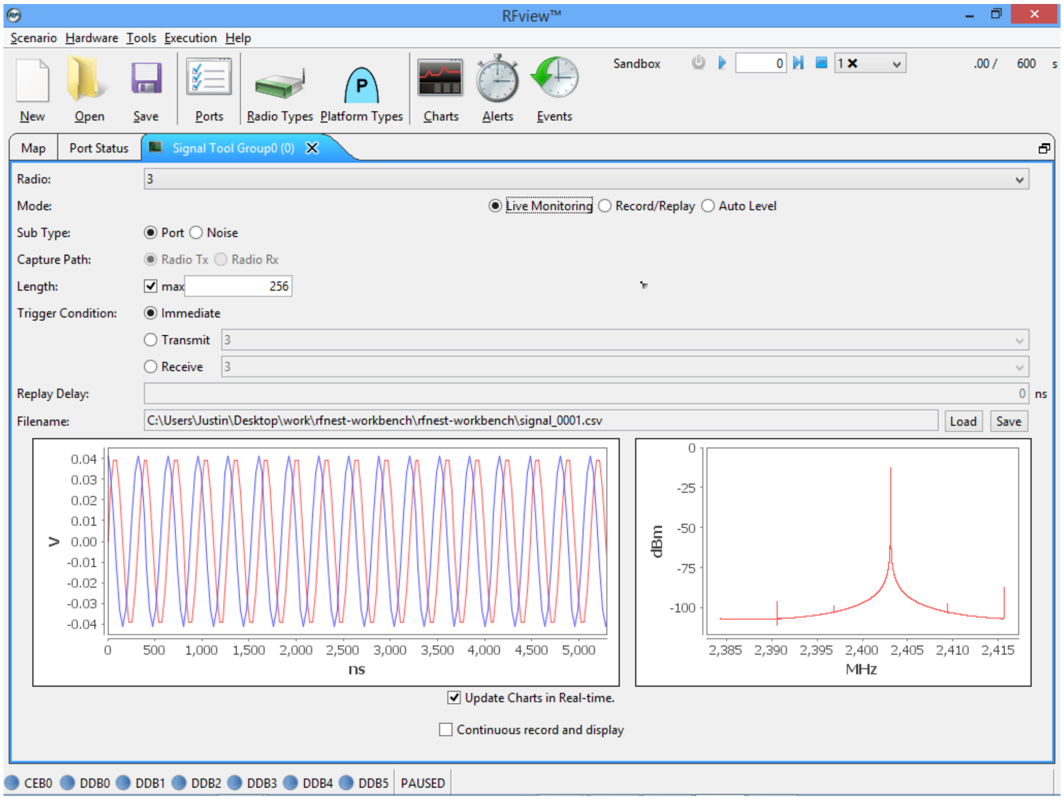
2. Select DDB which will be performing Record/Replay function.

   - DDB 0 has ports 0-3, DDB 1 has ports 4-7, . . . , DDB 5 has ports 20-23.

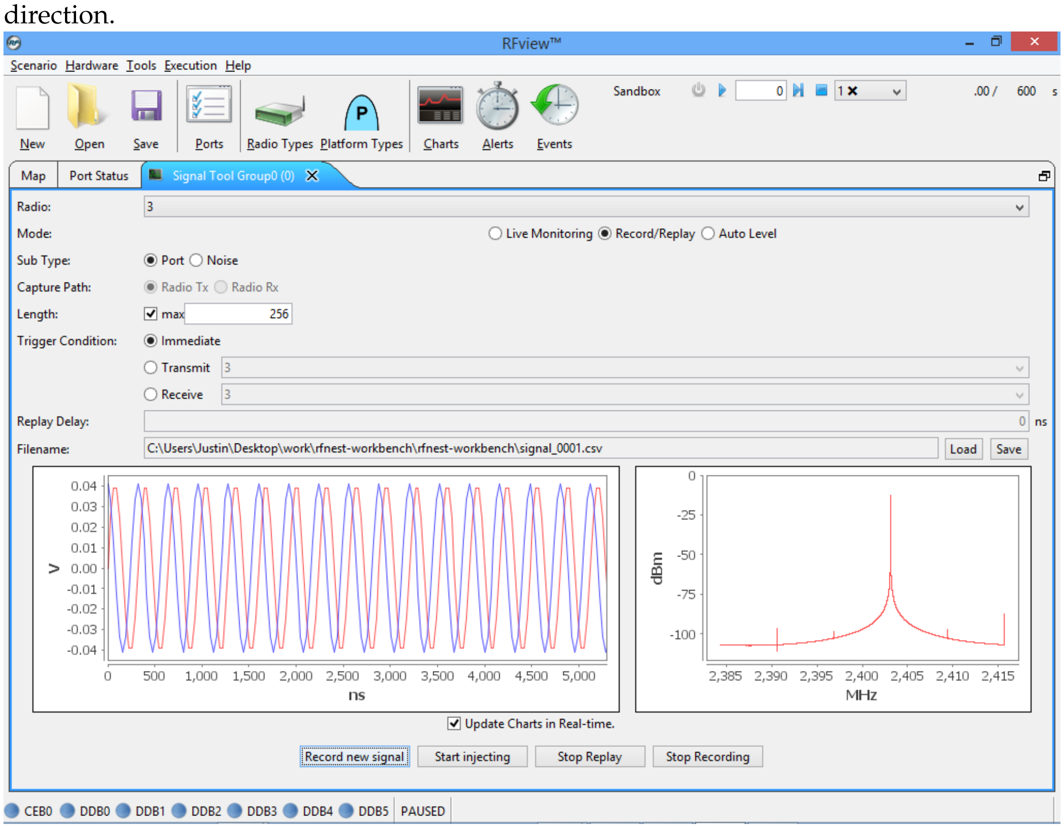
3. A 'Signal Record Group0 ("DDB#")' tab will appear next to the 'Map' and 'Port Status' tabs. Select the radio (port) to monitor or record/replay from using the 'Radio' drop-down menu.

4. Select 'Live Monitoring', 'Record/Replay', or 'Auto Level' using the 'Mode' radio buttons.

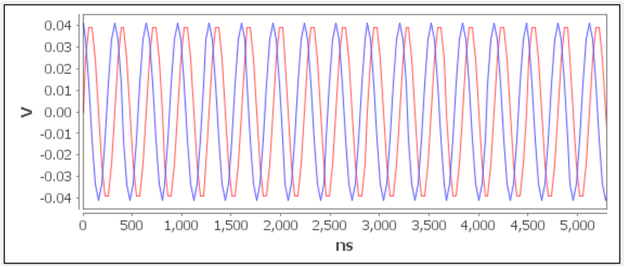
   - The 'Live Monitoring' option will continuously update the graphs with the signal is received from the hardware. User the "Continuous record and replay" checkbox to start/stop this behavior.

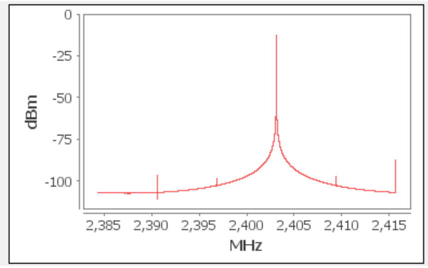


- The ‘Record/Replay’ option allows the user to initiate a signal recording or to stop/start a signal replay. Use the appropriate buttons on the bottom of the tab to ‘Record New Signal’, ‘Start Injecting’, ‘Stop Replay’, and ‘Stop Recording. Note that for each sub-type, only one action may be active at a time, and the previous action must be stopped before the next action is attempted. I.e., the user may replay a signal in the “noise” direction while also recording in the “port” direction.



- The 'Auto Level' option assists the user in automatically determining and calibrating the best input gain setting for each port. Before starting, select an estimated starting Input Gain for the current port via the Port Status tab. Next, ensure the radio is transmitting at least once per second. Finally, press the Auto Level button. The success or failure of auto-leveling will be displayed. If the initial starting Input Gain is significantly off, the signal is too weak or too strong to effectively sample, or the radio is not transmitting at a consistent power level, auto-level may fail. See Section 9.6.2 for more details.





5. The relevant loaded or recorded signal is displayed in separate time and frequency domain plots. The user can zoom in on the area of particular interest in the graph by clicking and dragging over a selected area on the graphs and can zoom out by dragging toward the top-left corner of the screen or by right clicking on the graph and selecting 'Zoom Out' and desired axis from the menu.

6. The following options are available for Live Monitoring or Record/Replay Mode:

   (a) Select the 'Sub Type' to record either the signal or noise on the port using the radio buttons.

   (b) Set the recording length in the 'Length' field or set it to maximum length using the 'max' check-box.

   (c) Set the 'Trigger Condition' to start recording using the radio buttons.

      i. When 'Immediate' condition is selected, Record/Replay will trigger immediately.

      ii. When 'Transmit' condition is selected, the Record/Replay will start on transmission from the radio selected from the 'Transmit' drop-down box. Note that if the radio's transmitted signal is too weak, the recording may never trigger.

      iii. When 'Receive' condition is selected, the Record/Replay will start on reception of the signal by the port set in the 'Receive' drop-down box.

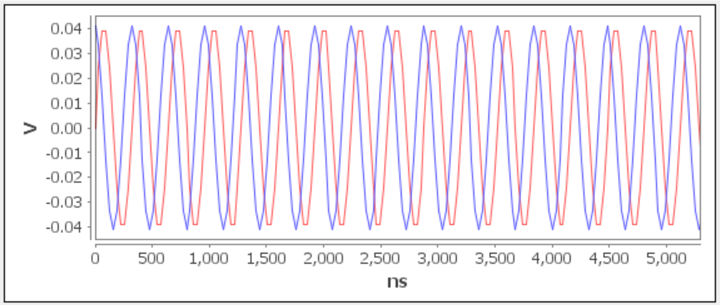
   (d) Set the delay with which the signal will be replayed in the 'Replay Delay' field.

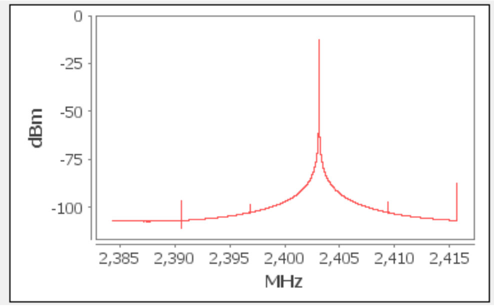
   (e) Set the path to save or load a signal in the 'Filename' field.

7. The "Update Charts in Real-time" toggles whether the graphs are updated when new samples are received from the hardware. **Note that if this checkbox is unchecked, the graphs will NEVER update.**

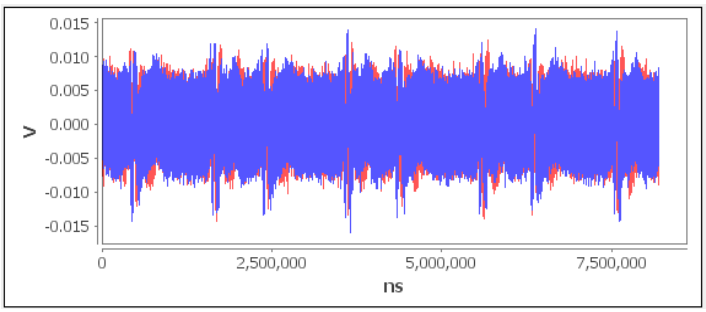
9.4.1 Examples of Recorded Signals

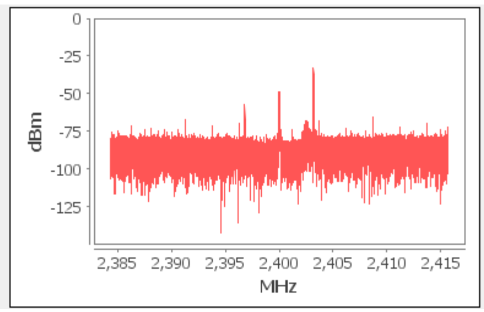
The signal below is a very clean signal indicating the input power and input gains are correctly set. Note that both the time and frequency domain. Note that the time domain signal has a smooth shape with little/no noise and the peaks of the signal are not clipped. The signal is typically not a perfect sine wave, but the time-domain plot should typically be smooth and without significant randomness or jagged edges, although some waveforms may always look noisy. The frequency domain shows a well-defined spectral shape which matches the expected shape of the waveform's transmissions. The frequency peak is significantly above the noise level of surrounding frequencies. A clean signal may not have as sharp of a peak. The more well-formed and less-ragged the frequency domain plot looks the cleaner the signal generally is, although some waveforms may always look noisy.





The signal below is a very poor quality signal resembling noise. Note that the time domain plot shows no definable shape and just a series of very jagged edges alternating between positive and negative amplitudes with little/no intermediate values. If zoomed in, the signal does not follow a typical sine wave, show typical modulation or pulse shaping characteristics, and is essentially indistinguishable from random noise. In the frequency domain, multiple small peaks are present and the noise is relatively high. Note that such a signal may result from the ADC being over-saturated, even if the power level is low. If such a signal is observed, either reduce the input gain or radio transmit power in 10 dB increments until the signal is no longer visible (proving the ADC is not over-saturated), or check for noise in the radio/cables/etc.





9.5 Attaching Radios

The radios must be attached to the system in a way which allows the RF signals to be digitally sampled from the transmitting radios at the desired power level, sent to the receiving radios at the desired power levels, and that leaking signals do not interfere with the system. Figure 1 shows the normal configuration for connecting two radios to the system in the case where the radios each have a single RF connector. Because each port on the hardware has a separate Rx and Tx connector, for receiving RF signals from the radio and transmitting signals to the radio, respectively, a power splitter must be used. When using a splitter, ensure that the “S” or input connector on the splitter is attached to the radio. If the radio has separate Rx and Tx connectors, connect the **Tx** connector on the radio to the **Rx** connector on the hardware and the **Rx** connector on the radio to the **Tx** connector on the hardware. Note also that the top connector on each port is the Rx connector and the bottom connector is the Tx connector.

When attaching radios, there are several concerns that must be addressed simultaneously and properly configured into the system.

> *NOTE: Prior to following the instructions in this section to verify the radios are attached correctly, the system must first be successfully initialized.*

- **Appropriate transmit power** – The transmit power of the radio should be as low as possible, typically around 0 dBm although some radios may transmit at 26 dBm (500 mW) on their lowest setting. After the transmit power of the radio is lowered, one or more attenuators must be placed on the radio to attenuate the signal before receipt by the hardware.
- **Appropriate over the air isolation/shielding** – The radios will leak a signal over the air between each

other, effectively bypassing the emulated channel created by the system. Typically a consumer grade radio is not designed with shielding in mind and may only provide 80-90 dB of isolation if radios are adjacent to each other, requiring additional shielding to be used. A commercial or tactical radio may or may not have much better shielding. Apart from the radios leaking directly, the cables or especially the attenuators used may create leakage if not shielded or spaced properly. In general, try to avoid placing the attenuators and cables of different radios next to each other, for example by placing the attenuator as close to the radio as possible.

- **Appropriate power and attenuation for dynamic range and system leakage –** The hardware itself provides between 60 and 95 dB of isolation, depending on the frequency and port pair in question. This isolation is measured from the Rx (and Tx!) connector on one port to the Tx connector on another port. This issue is explained in more detail in the following sub-sections.
- **Appropriate gain and transmit power offset configuration –** The system must be configured with appropriate input and output gains for the radios and scenario to function. The system must also be aware of the difference between the radio's current transmit power and the power the radio uses in the field as well as the amount of attenuation applied externally to signals generated by the hardware. This configuration is also explained in more detail in the following sub-sections.

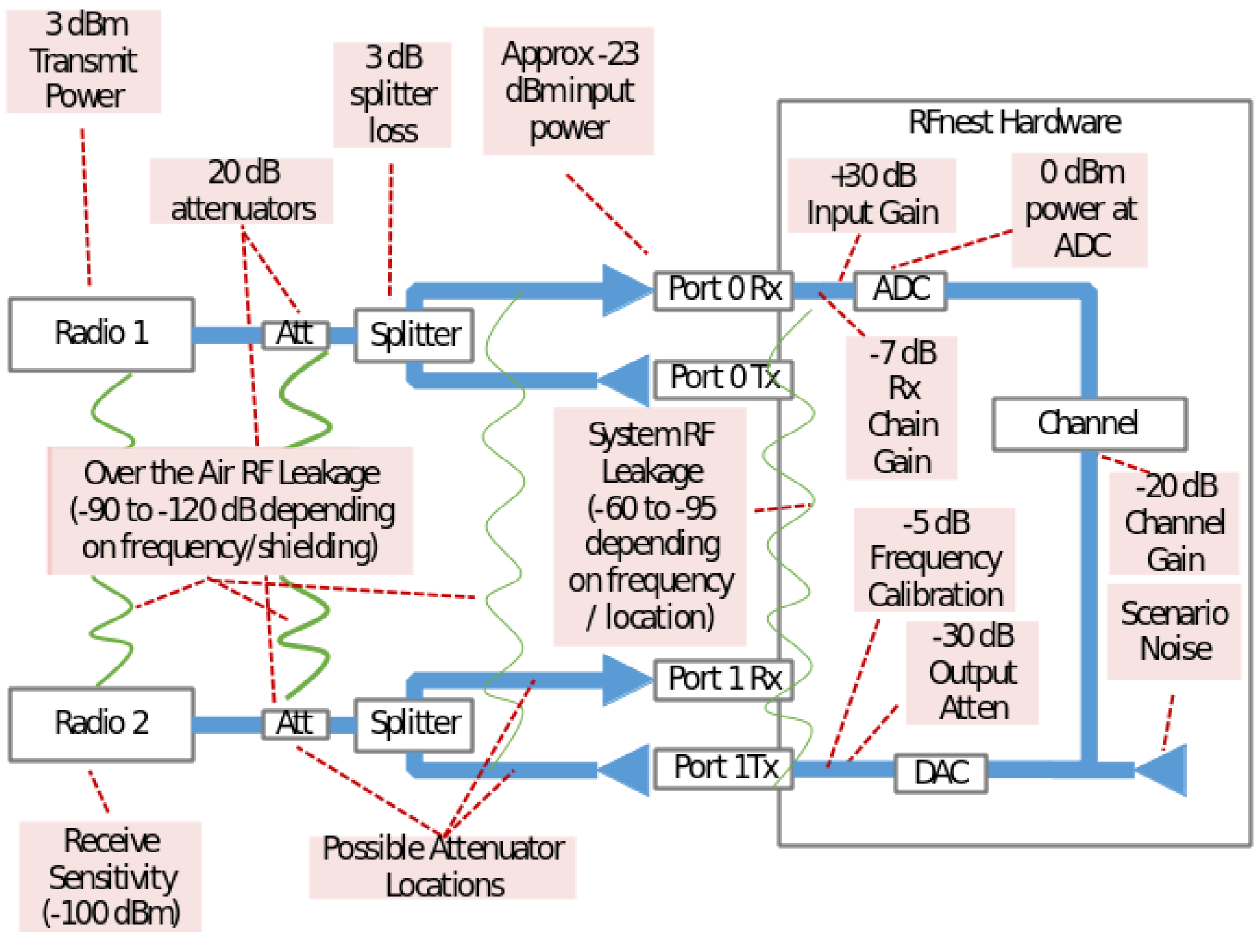


Figure 1: Understanding cabling, gains, and leakage in the D-Series Hardware

NOTE: For the A-Series, a splitter is not required if the radio attached has a single, combined Tx/Rx connector.

9.6 Choosing External Attenuation, Setting Gains, and Calibrating Radios

The next step in correctly operating radios with the system is to achieve the needed input power on the Rx ports and to determine the desired dynamic range of the system based on the receive sensitivity of the radios under test and the intended scenario.

9.6.1 Determining if the radios will be Isolated

Continuing with the example in Figure 1, the input power reaching the hardware should be between -10 dBm and -40 dBm, with around -20 dBm the preferred value. In this example, the radio transmits at 0 dBm and thus has a 20 dB attenuator followed by a 3 dB loss due to the splitter, making the input power to the system approximately -23 dBm. If the radio's transmit power setting is too strong, leakage may occur, preventing links in the scenario from being broken. Assuming the radio's sensitivity is -100 dBm and the system leakage is -60 dB, this "leaked" signal will be 0 dBm minus 20 dB for the attenuator at the radio minus 3 dB for the splitter at the radio minus 60 dB system leakage minus 20 dB for the attenuator at the receiving radio minus 3 dB more for the receiving radio's splitter and the combined cable loses, making the leaked power through the system -106 dBm, below the receiving radio's sensitivity. Similarly, assuming the radios, cables, and attenuators are isolated by more than 100 dB, the over the air leakage will not exceed the -100 dBm sensitivity of the receiving radio. These measurements and calculations serve to confirm that isolation is achievable (which will be verified in the next sub-section) and are not input into RFview.

9.6.2 Ensuring the input signal is at the correct level (D-Series Only)

At the same time, the ADC which digitizes the RF signal as it enters the hardware may also become saturated or may receive too weak a signal, preventing the radio from communicating successfully in the scenario. The Rx port has a controllable gain stage to address this. In this example, the Rx port receives an input signal at -23 dBm and also experiences -7 dB of losses in the hardware Rx chain at this frequency, making the signal at the ADC -30 dBm, while the desired level is 0 dBm. With this in mind, go to the Port Status widget and select the port that Radio 1 is attached to. If unsure of the port, refer to the Port Configuration Summary dialog. Under the Properties tab, enter 30 (dB) as the Input Gain. The new gain will be applied to the system within 1-2 seconds.

To verify the input signal is at the correct level, the radio must be transmitting at least once per second. It does not matter if the waveform is such the radio continuously transmits (e.g. OFDM) or only transmits randomly (e.g. TDMA or CSMA), as long as the radio transmits at least once per second. Now, go to the Signal Tool for the DDB containing the port in question (DDB0 contains ports 0-3, etc.). The Signal Tool can be found by right-clicking on the radio's group in the group list or in the menu under Tools, Views, Signal Tool. Select the radio and select "Auto Level" under mode. Press the "Auto Level" button at the bottom of the Signal Tool. The radio's transmission will be repeatedly captured and displayed. The Signal Tool will attempt to automatically fine-tune the Input Gain so that a 0 dBm input power at the ADC is achieved. For most signals, 0 dBm is achieved when the amplitude of the peaks in the time domain (left) plot are at approximately ± 0.17 V. In some cases, auto-leveling will take up to 1 minute and/or may report failure even though the signal amplitude appears to be at ± 0.17 V. This may occur if the signal power fluctuates, since the auto-level function requires three consecutive measurements at or near 0 dBm before success is reported.

See Section 9.4.1 for examples of good and bad signals. If no signal appears or the signal is well below 0.17 V and failure is reported, in most cases where this occurs the input signal should be made stronger. In the Port Status widget, confirm that the input gain is set to 38, the maximum value. This indicates that the signal is too weak even with the maximum input gain applied.

Similarly, in some cases, the input gain entered by the user may be too strong, or the input signal may be too strong, or both. If auto-leveling fails and signal amplitude is significantly above 0.17 V, check the input gain in the Port Status widget. If the value is -9, the minimum value, then the input signal is too strong for the system to accommodate and should be reduced.

In other cases, the auto-leveling may incorrectly succeed. For example if the radio is generating a very strong input signal and the input gain is set very high, the ADC will saturate and essentially random values will be sampled. This will confuse the auto-leveling. To ensure this did not occur, look at the frequency domain (right) plot. If the shape of the spectrum does not match that expected and/or is very flat and "hairy" in appearance, the ADC may be saturated. Enter an input gain that is 10-20 dB lower and re-run the auto-leveling.

NOTE: At times, a 0 dBm input power may be unachievable. If this is the case, the desired power used by the auto-level function can be adjusted. This value is found in the Group Configuration dialog under the Calibration tab for that radio's frequency group. Once adjusted, auto-level can be run and will instead try to achieve the new power level. In general, the user should attempt to achieve a 0 dBm input power at the ADC.

Ultimately, it is up to the user to decide whether the input gain is set correctly and fine-tune the value manually using the Port Status widget as appropriate. Note that the success/failure of the auto-leveling is only for the user's benefit, the system may still function normally if auto-leveling reports failure but the signal's input power and the configured input gain are observed to be correct. The signal can be manually recorded and viewed by changing the mode in the signal tool to "Record/Replay" and pressing the "Record new signal" button. It may be useful to maximize the Signal Tool widget or zoom in on the plots to view the signal more clearly.

NOTE: Due to variations in radio behavior, losses in various RF cables and splitters, and other factors, this step must be done for each radio connected to the system.

9.6.3 Adjusting the Transmit Power Offset

The Transmit Power Offset, TPO, tells the system how much of the gain applied to a channel is done digitally versus has already been applied via the radio transmitting at a different power level than used operationally, external attenuators, a non-ideal input power level, and the output gain. There are two ways to approach the calculation of the TPO, using the radio's sensitivity and desired maximum SNR, or by simply adding up these factors. In theory, both approaches reach the same result.

Note that the TPO can only be adjusted per frequency group.

NOTE: The TPO is only applied to the system on initialization of the system. Each time a setting is changed which affects the TPO, stop and re-initialize the system.

9.6.3.1 Dynamic range method

To determine the TPO using the desired dynamic range, the radio sensitivity and desired maximum SNR must be considered. Again in this example assume the radio's sensitivity is -100 dBm. The radio must then be able to experience signals weaker than -100 dBm, for example -110 dBm, in order to "break" links used by the channel. Similarly, the link must also be able to be "very strong" i.e., have a high SNR. Assuming the noise floor is -95 dBm and the waveform operates at maximum performance under a 30 dB SNR, a strong link must also be able to exceed -65 dBm, for example -60 dBm. The desired power range at the receiving radio is then -60 dBm to -110 dBm. Note that in this example Radio 2 is treated as the receiver for

simplicity, however the situation is symmetric, Radio 2 must also be able to transmit to Radio 1. Since a 0 dBm power level at the ADC was achieved, the system can digitally output a power level between 0 dBm and -60 dBm. The TPO is then selected to shift this range into the desired dynamic range. For example, if -50 is the TPO, the system will generate output between -50 and -110 dBm.

The TPO cannot be directly entered, instead it is calculated based on several parameters. The User Transmit Power Offset should equal the attenuation applied between the Tx port Radio 2 is attached to and Radio 2, in this case 20 dB. Next, the hardware's output attenuation can be controlled in 10 dB increments. For example, select -30 dB. The frequency calibration is automatically determined based on the system calibration. For example, 5 dB may appear based on the current frequency and output attenuation. As a result, the TPO will be calculated at 20 - (-30) + 2 = 55. Since that TPO will achieve a dynamic range of -55 to -115, this may be acceptable. If not, either adjust the output attenuation value, or change the external attenuation and adjust the user transmit power offset accordingly. Note that the user transmit power offset should not be updated without changing the external attenuation.

9.6.3.2 Scenario based method

The alternate way to view the the TPO is a values which tells the system how much of the gain on a given link to be applied digitally versus has already been applied via other means. The TPO is then simply based on the difference between the operational transmit power of the radio and the input power at the ADC, plus the internal and/or external attenuation that the receiving radio has between the radio and the hardware's Tx connector for that radio. This attenuator is essentially equivalent to further lowering the radio's transmit power. In this example, if the radio normally transmits at 25 dBm and the receiving radio has 20 dB attenuation, the user transmit power offset should be entered as 25 dBm (the operational transmit power) - 0 dBm (the ADC input power achieved) + 20 dB (external attenuation) = 45 dB. As with the previous method, the 5 dB frequency calibration is added giving a final TPO of 50. When the scenario dictates a gain to be applied to the channel between these radios, the system will then add this TPO from the channel gain that is digitally applied.

For example, if -70 dB is the gain specified in the scenario (e.g., based on node positions), -70 + 50 = -20 dB will be used as the channel gain to be digitally applied. The result will be that the receiving radio will receive a signal that is 70 dB weaker than the radio's operational transmit power (0 dBm ADC input power - 20 dB digital channel gain - 5 dB frequency calibration – 20 dB external attenuator = -45 dBm, while the radio operationally transmits at +25 dBm). In other words, 50 dB of the desired loss has already been applied when the scenario was created and radios were attached, leaving only 20 dB left for the system to digitally apply to reach the -70 dB gain dictated by the scenario. This method is more correct from a scenario standpoint, but may create a dynamic range which never breaks the links between radios. If this occurs, increase the output gain setting until isolation is achieved as described in the next sub-section.

9.7 Verifying Radio Operation and Isolation

Once the radios are attached and the system is configured via the instructions in the previous sub-sections, the operation and isolation of the radios must be verified. This step verifies that when two radios have a "very strong" link they can communicate while when they have a "very weak" link they cannot. This step requires two radios to be present, since the step involves verifying that one radio can (and can't) transmit to another via the system.

9.7.1 Verifying Radio Operation

First, to verify the radios' operation, attempt to transmit a signal from one radio to another. Under the Links tab, select the two radios. Next, view the Channel Properties Editor tab. The link should be displayed.

Verify that the port numbers match those of the two radios being verified. Select "manual" and adjust the gain to 0 dB. This action causes the channel between these two radios to be the strongest possible. Note that some loss is still occurring (caused by the attenuators, splitters, etc., i.e. a loss equal to the TPO) but no digital gain is applied once the gain is below the TPO. In other words, if the TPO is 50, once the gain is set to a value above -50 dB, the digital gain will be set to maximum (0 dB).

Via the radios' display or other means (ping, wireshark, tcpdump, etc.) that the radios are able to communicate. It may be useful to independently check each direction (i.e., radio 1 transmitting to radio 2, followed by radio 2 transmitting to radio 1) to help narrow down any problems that occur.

If the radios cannot communicate, there are two possible causes. For the D-Series, the input power at the ADC may be too high or too low. To adjust this refer to Section 9.6.2. Alternatively, the losses between the Tx port of the receiving radio and the radio may be too high. Reduce the output gain, decrease the external attenuation present, or choose cables with less loss.

To verify the output signal is reaching the receiving radio and determine its power, a spectrum analyzer can be used. Attach the spectrum analyzer in place of the receiving radio or split off the signal using an additional splitter. Tune the spectrum analyzer to the frequency used by the radios and observe the power level. If the radios do not transmit continuously, it may be useful to use the "Max Hold" function of the spectrum analyzer to assist in measuring the power when the radios are transmitting. This function is usually available under the "Trace" menu on the spectrum analyzer.

If a spectrum analyzer is not available, in the D-Series the signal can also be viewed digitally within the system prior to passing through the DAC and the internal attenuator on its way out the Tx connector. To do this, use the Signal Tool and select the receiving radio. Select Record/Replay for the mode and "Noise" for the sub-type. Under Trigger Condition, select "Receive." There is no need to choose a radio in the drop-down next to receive as this is ignored for recordings. Now press "Record new signal." This instructs the system to record the digital signal about to leave the Tx port, and to start recording only when a signal is present. If the signal is blank or only a straight line is shown, the transmitting radio may not be transmitting. Try pressing the "Record new signal" button several times, eventually a signal may be captured. Once captured, observe the power level in the plots. The plots should show the same signal as that captured when the transmitting radio's input signal was being verified to be at the correct power level at the ADC. In the frequency domain (right) plot, the spectral shape should be clean and match that expected. Assuming cables are attached correctly, the cables haven't failed (this is not unheard of), and the internal and external attenuation don't cause the signal to have a SNR below that required by the radio, communication should work.

The path starting from the DAC cannot in general be verified by the system alone. In cases where the signal is strong enough and/or the transmitting radio is transmitting continuously, the cable entering the receiving radio can instead be attached to another **Rx** port on the system, allowing the final signal level to be measured by the system itself. Note that the input gain of the system needs to be considered if this technique is used.

9.7.2 Verifying Radio Isolation

After the radios are able to communicate, it must be confirmed that the radios are actually communicating via the system, and not via the leakage paths shown in Figure 1. In other words, the radios may not be communicating via the system at all (they may only be communicating via leakage) meaning that the radio operation has not been verified. Alternatively, the radios may be communicating via the system when the channel gain is high, but when the channel gain is low the radios may still be able to communicate via the leakage and thus preventing the link from being broken correctly by the scenario.

To verify isolation, simply return to the Channel Properties Editor with the link between the two radios still selected. Now, adjust the gain to -200. Unless the TPO is set to an extremely high value (i.e., above 104), this will cause the system to digitally break the channel. **Absolutely no** signal will reach the receiving

radio via the channel. After setting the gain to -200, attempt to get the two radios to communicate. If the radios can still communicate, then they are communicating via leakage.

To eliminate system leakage as a possibility, disconnect the RF cables for one or both of the radios from the system but leave the cables in approximately the same position as if they were connected. If the radios can still communicate, the communication is happening via over the air leakage. Otherwise, they are communicating via system leakage. In this case, verify the input power at the Rx connector on the transmitting radio's port. If this power level minus the system isolation (worst case is 60 dB) minus the lesser of the attenuation/loss between the Rx or Tx connector and the receiving radio is greater than the receive sensitivity of the radio, the system leakage will pose a problem. Note that the system leakage may occur via **either** the Rx or Tx connector of the port the receiving radio is attached to. Note also that the system leakage is **not** affected by the input gain or output attenuation/gain settings on the system.

If the radios are communicating via over the air leakage, either 1) the radios' transmit power must be lowered; 2) better shielding must be placed around the radios and their attenuations/splliters/etc.; 3) better shielded (i.e. double-shielded cables) must be used, or 4) the radios and these other sources of leakage must be moved farther apart from each other. Assuming well-shielded cables are used and all external attenuators are placed near the radios, use 10' or longer cables and move the radios in opposite directions until they are 20' feet apart and separated by as many RF-blocking/absorbing objects as possible. Thick walls, filing cabinets, other metallic objects, etc. may serve this purpose. If the radios stop communicating when the radios/attenuators/etc. are moved far apart, over the air leakage is confirmed. Since large distances may not be practical, lowering the transmit power of the radio and/or reducing the RF power (by attenuating as soon as possible) is key. Commercially available RF shielding boxes can also be used, as can copper mesh, RF absorbing foam, etc.

9.8 Configuring Thresholds for Scenario Layout/Display

Now that the radios are successfully connected to the system, the scenario can be fully laid out and tested. RFview provides several features that facilitate the design of scenarios, the analysis of network performance/behavior, and create more impactful demonstrations. These features are the Gain vs. Distance Overlay and the Link Quality Overlay. **When properly tuned, they allow the user to clearly visualize the state of the links and the underlying cause behind various behaviors.** The Gain vs. Distance overlay shows circular regions in which a selected radio's links are in a good, fair, poor, or bad state. The Link Quality Overlay allows the quality of each link in the entire scenario to be visualized using either the link's configured gain and/or the radio's transmit power, or the RSSI reported directly by the radio itself.

The benefit if these tools is only realized when the thresholds are tuned properly. First, decide whether the expected (based on the link's configured gain) or observed (based on reported RSSI, see Section 10.1. The observed gain offers superior accuracy but may not be readily available from the radio and can be subject to fluctuations/misreporting/unavailability due to radio quirks and EW aspects of the scenario. The expected gain is always available, since the values are obtained from the CEC. In theory, the observed and expected gains are equal. If they differ significantly, the TPO may be incorrectly set or the radio may be misreporting the RSSI.

> *NOTE: The gain to use for drawing the links on the map must be chosen and enabled by the user. This is done in the Link Quality Overlay. If in doubt, enable the display of the Expected gains.*

Tuning these tools is done via the Gain vs. Distance Overlay's configuration widget. Select the Gain vs. Distance Overlay in the overlay list, select a platform, select a radio within the platform, and then click on the Edit button. Pressing Edit allows all sub-types to be edited, so selecting one in advance is not needed. Now, enter the signal levels above which the link is considered to be good (green), fair (yellow), and poor (red). Note that the colors in parenthesis are used by default and can be changed via the Link Quality Overlay's configuration widget. Any link below the poor threshold is considered broken and is not drawn on the map.

If the signal level thresholds at which the link becomes good, fair, and poor are not known, they must be selected by the user. The definition of good, fair, and poor are themselves up to the interpretation of the user based on the scenario. For example, a link may be considered good if it allows the maximum data rate with no loss. Alternatively, good may be considered to be the threshold at which a given video stream can be played over the link with "good" quality.

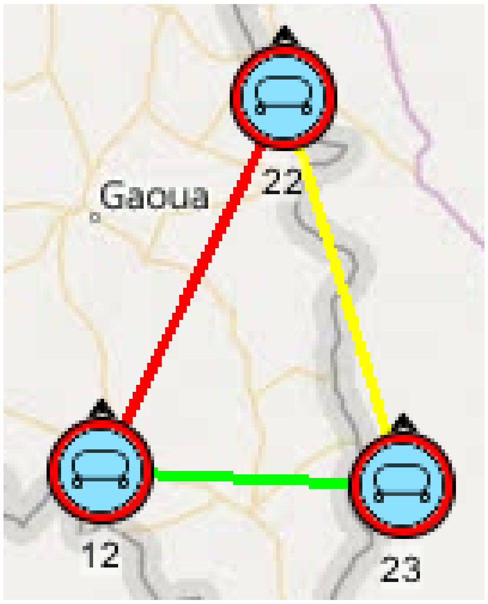
To determine the thresholds if unknown, first start the measuring technique to be used. This may be a video application, iperf, etc. Now use the Channel Property Editor to adjust the link between two radios to various levels. Write down the values when the link quality changes based on the user's judgment. Now, enter these values into the Edit dialog in the Gain vs. Distance Overlay for that radio's frequency group and sub-type (select the platform containing the radio, then the radio's frequency group, then the sub-type). After saving the new values, change the Channel Property Editor back to automatic mode for that link, allowing the system to automatically calculate these values based on distance. Now, move the two radios various distances apart. The perceived link quality should change in conjunction with the radios moving outside each other's good/fair/poor circles and the link changing color. If not, re-adjust the thresholds. To precisely adjust the gain of a particular channel during this process, the Channel Properties Editor tab can be used in manual mode. Using the Channel Properties Editor, the link color will update however the circles drawn will not correspond any longer since manual setting, rather than distance-based models, are being used to calculate that link. Repeat this process for each frequency group and sub-type in use by the scenario. While somewhat time-consuming initially, this step saves considerable time during scenario layout, results analysis, and improves the understandability of demos.

> *Note: If the link never works or never breaks, or only works/breaks at the extreme edges of the Gain vs. Distance graph, the Transmit Power Offset and/or Output Attenuation should be adjusted as described previously in this section.*

10. Basic Monitoring and Data Collection

10.1 RSSI/Link Quality

RFnest™ provides the ability to visualize the RSSI value or other link quality metrics in real-time for links between nodes in the scenario. The format is described in the RFnest™ API document. This capability allows the user to see the links between radios/nodes, to help see the scenario and understand the resulting network behaviors. The link quality is collected by the user or system under test and specified via the RFnest API on a per-link basis. When the "observed" link quality is selected via the Link Quality Overlay, the links between nodes are then colored based on the colors configured in the Link Quality Overlay configuration and the thresholds configured in the Gain vs. Distance Overlay configuration.



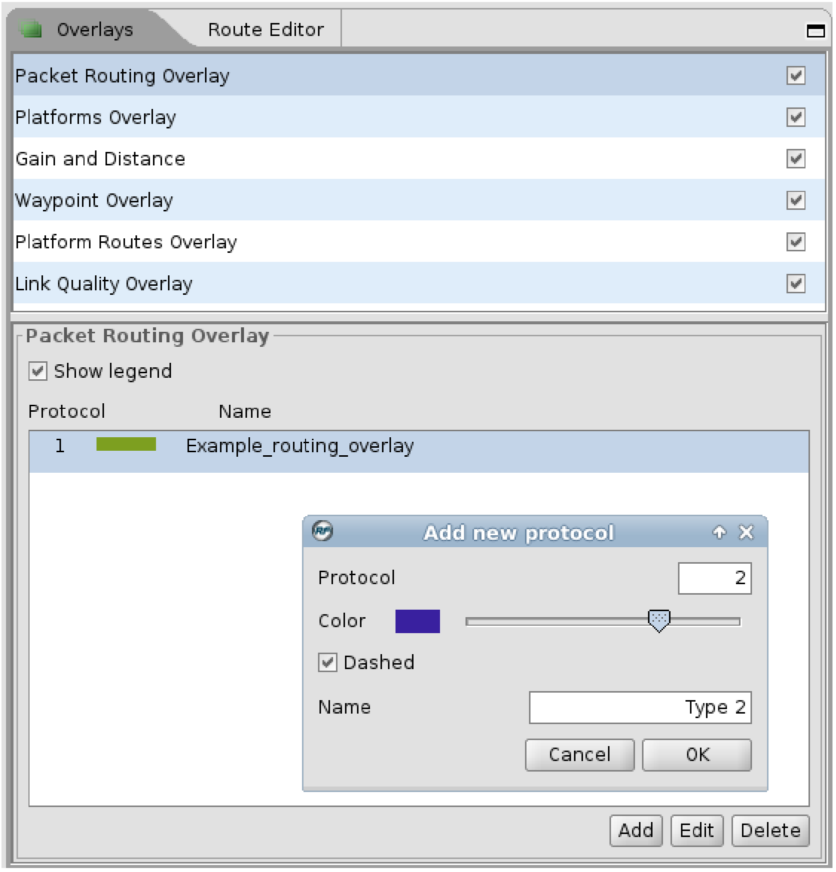
10.2 Routing/Communication Events

Routing or other communication information/events can also be visualized by RFnest™ using arrows displayed between nodes.

10.2.1 Packet Routing Overlay

This configuration allows the user to specify the color for routing overlay per node.

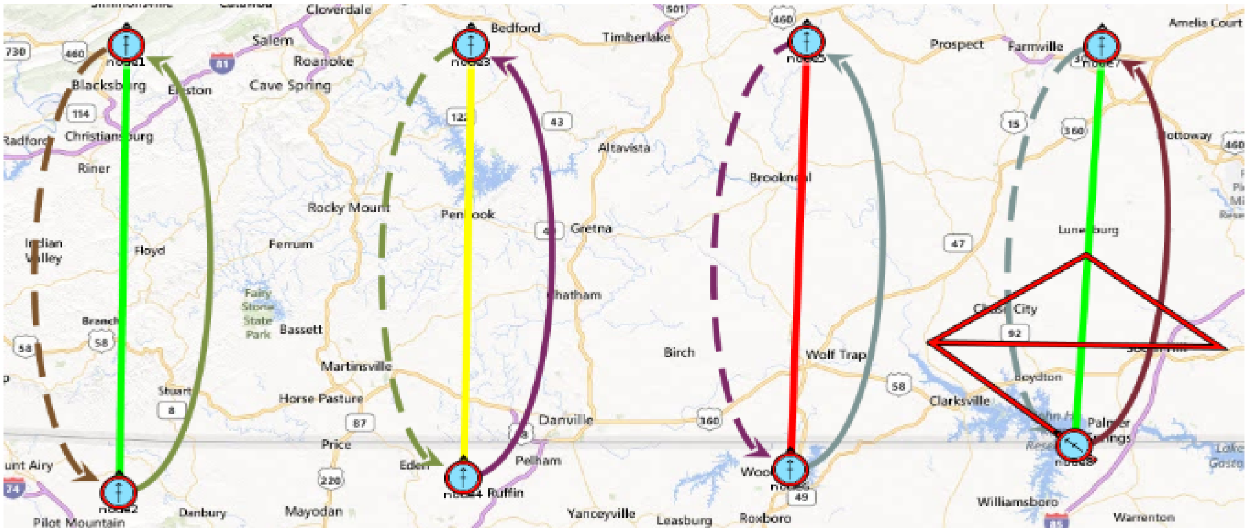
1. In the 'Overlays' tab check or uncheck the 'Packet Routing Overlay' checkbox to show or hide the routing arrows.
2. Click on the 'Packet Routing Overlay' field to show the routing overlay editor.
3. Click 'Add' to add new protocol.



4. Set the 'Protocol' number.
5. Set the color by dragging the 'Color' slider.
6. Set the routing arrow to be dashed or solid by checking or unchecking the 'Dashed' checkbox.
7. Set the name.
8. Click 'OK' to save changes, or 'Cancel' to cancel.
9. To show routing arrows in RFview, the user can send periodic updates to RFview's multicast address: 224.1.2.208 on port 20861 using the format specified in the API.
10. The routes will stay on the map for 1 second and then disappear unless another update is sent before then.
11. If a route update is sent for the same source node and protocol number but a different destination, the existing arrow will be immediately replaced with a new arrow pointing to the new destination.

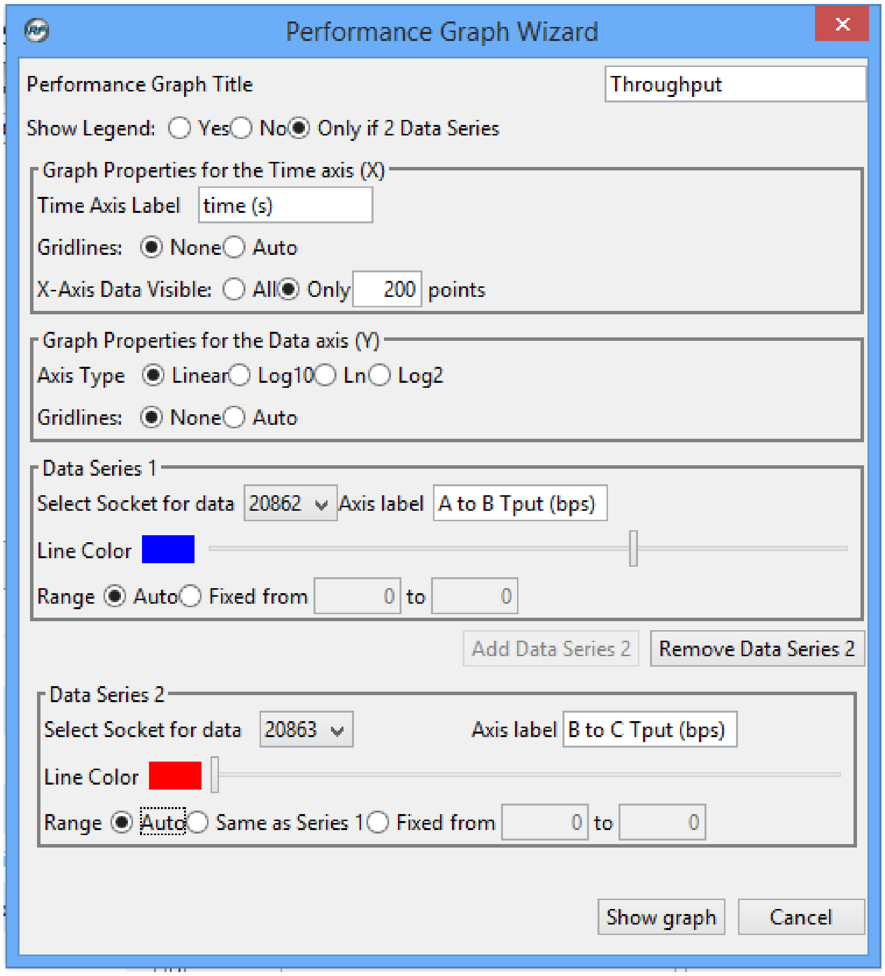
10.2.2 Usage in a Scenario

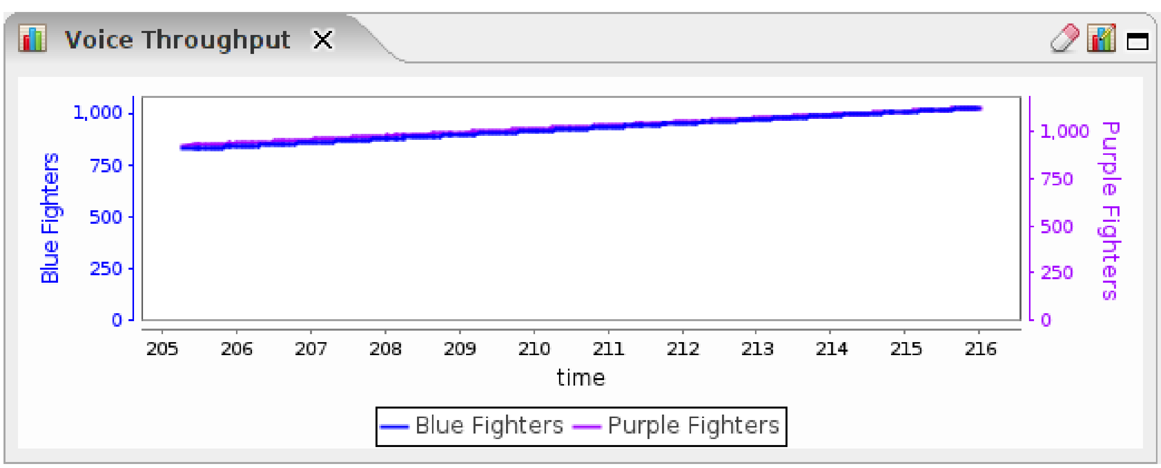
As an example, arrows can be drawn in real time to show the path that packets are taking between two nodes. To configure, click on the routing legend in the upper-right corner of the map. This menu allows the name, color and other properties for each "protocol" to be defined and for protocols to be added/deleted. The format for these updates is described in the RFnest™ API document, where the protocol specified in the updates corresponds to the index (e.g., 0, 1, 2, 3, etc.) and their properties as configured by the user. Note that the arrows are automatically removed after a period of time unless additional updates are sent to prevent them from expiring. This expiration prevents the process generating the messages sent to RFview from having to maintain complicated state.



10.3 Real-time Graphs

Real-time metrics, such as the throughput between two nodes, can be recorded and displayed by RFview. To send data points to the graph, updates can be sent to the specified port as described in the RFnest API document. As points are added, old points are automatically removed. Note that the automatic scaling of the X axis (to allow data points to be sent at arbitrary intervals) can cause the graph to be temporarily stretched/compressed, if data points are sent at highly irregular intervals.





11. Advanced Analysis Tools and Topics

11.1 Adding Point Source Emitters and Background Noise (D-Series Only)

RFview supports the digital addition of additional RF sources via two methods, point source emitters and

background noise. The difference between the two is that a point source emitter is a signal generated from a particular location and travels through the channels defined by a radio/platform's location in the scenario while the level of a background noise signal bypasses the channels and instead uses only a configurable on a per port basis.

11.1.1 Point Source Emitters

To add a point source emitter, configure a platform and radio exactly as if a real radio at the desired frequency was being attached. Assign the new radio to a port via the Group Configuration dialog. Note that the new radio must be configured in this dialog to have a RF frequency equal to that of at least one other existing radio, otherwise the new radio will be a member of its own frequency group and thus no signal will go through the channels.

Position the emitter's platform in the environment and, if desired, define the mobility of the platform. The emitter's channels to other radios in the same frequency group will be calculated automatically.

Use the Signal Tool to select the emitter's radio. Select Record/Replay mode. Select the "port" sub-type. Click on the load button to load a previously saved or custom generated (e.g. in Matlab) signal file. Press the Start Injecting button. The loaded signal will now be played exactly as if the signal were being received by the RFnest hardware from an RF source attached to that port.

The Events Editor can be used to trigger the start, stop, and changing of the signal generated by the emitter automatically at specific times in the scenario.

11.1.2 Background Noise

To add background noise, go to the Signal Tool of the DDB for the port(s) where the noise is to be generated. Note that if noise is to be generated on all ports, these steps must be repeated for each DDB.

Select Record/Replay mode. Select the "noise" sub-type. It is not necessary to select a radio, as the signal will be replayed on **ALL** of that DDB's ports/radios. Click on the load button to load a previously saved or custom generated (e.g. in Matlab) signal file. Press the Start Injecting button. The loaded signal will now be received by all radios attached to that DDB. To adjust the gain with which each radio attached to that DDB receives the signal, go to the Port Status Tab under the Properties section for each of the ports of the DDB. To make the port receive the background noise signal exactly as loaded, set the noise gain field to 0 dB. To make the port receive NO noise gain signal (i.e., selectively disable the noise for individual ports) set the noise gain field to -100 dB. Intermediate values can be used to various gain levels. Note that the noise signal gain is NOT affected by the TPO setting for the frequency group or the channel properties in any way, but is affected by the output gain for each port (i.e., the output attenuation set in the Group Configuration).

The Events Editor can be used to trigger the start, stop, and changing of the background noise signal generated by each DDB automatically at specific times in the scenario. Note that the noise gains cannot be configured to be changed via the Events Editor.

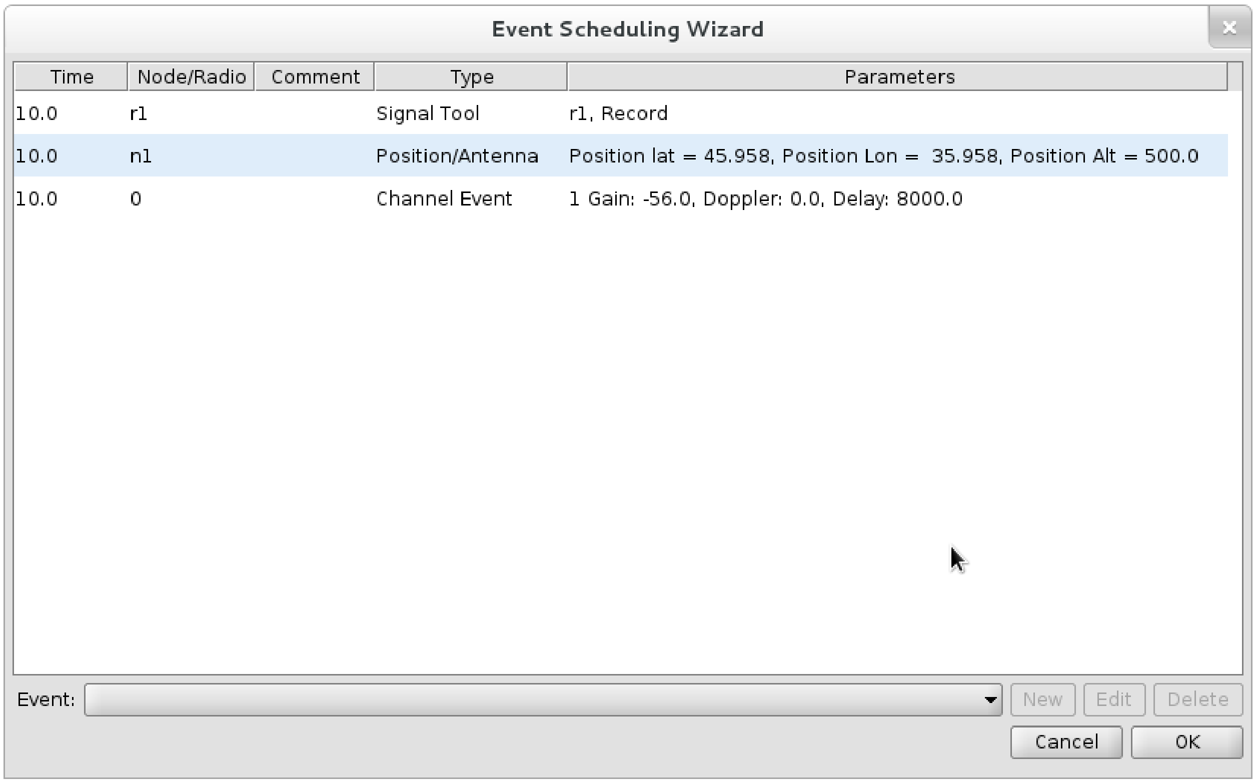
11.2 Events Editor

Events are used in scenarios to either Record snippets of radio inputs or Play recorded snippets of radio inputs. The standard use for events are for either making recordings at a specified time (for later comparison between runs/radios), or recording and then replaying signals to emulate EW. Alternatively, this tool can also trigger a manual record or manual replay event in the scenario as well.

11.2.1 Create event

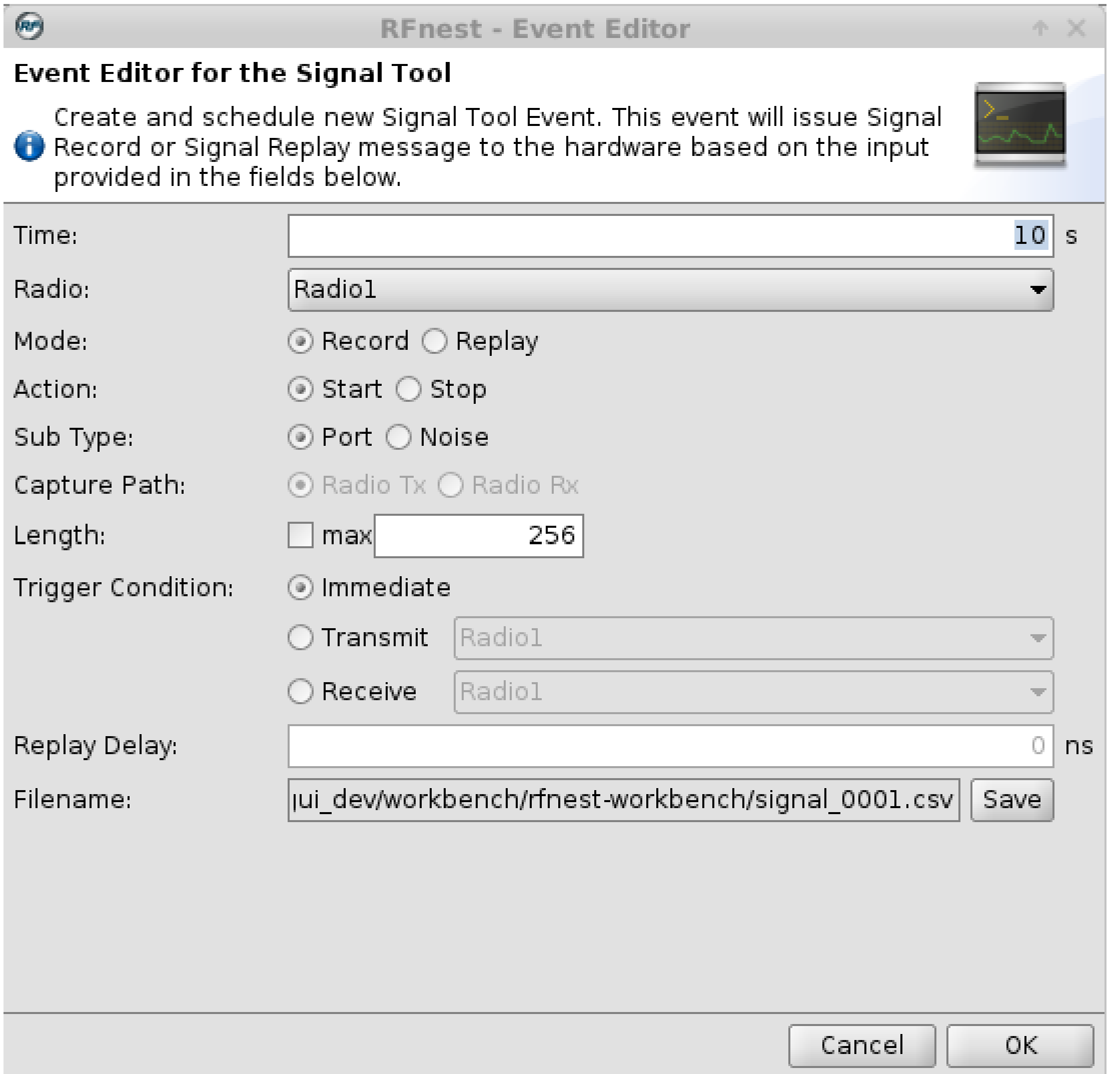
1. Open the 'Event Scheduling Wizard' by going to the 'Tools' menu and selecting 'Events' or clicking on the 'Events' icon on the toolbar.

2. Select the type of event from the 'Event' drop-down menu and click on 'New' button to create a new event.



11.2.2 Create Signal Tool Event (D-Series Only)

1. Select 'Signal Tool Event' from the 'Event' drop-down menu and click 'New'.



2. Set the time when of the event execution during the scenario in the 'Time' field. If 0 is set for the 'Time" field, the event will fire when the scenario starts, i.e., at 0.0 time.

3. Select the radio (port) which will record or replay a signal from the 'Radio' drop-down menu.

   - Note that the selected port will replay a recorded signal only, and if there is a radio connected to the port its signals will not be going through the system why replay is active.

4. Select 'Record' to trigger recording of a signal or 'Replay' to trigger replaying of a signal by selecting the appropriate 'Mode' radio button.

5. Select 'Start' or 'Stop' radio button from the 'Action' field to start or stop the record or replay event.

6. Select type of signal to record from the 'Sub Type' radio buttons 'Port' and 'Noise'.

   - When 'Port' is selected, the signal incoming on that port will be recorded.
   - When 'Noise' is selected, only the background noise and not the incoming signal on the port will be recorded.

7. Set record or replay length in the 'Length' field or check the 'max' checkbox to set the record/replay time to maximum.

8. Select the condition to trigger record/replay by selecting one of the 'Trigger Conditions': 'Immediate', 'Transmit', or 'Receive'.

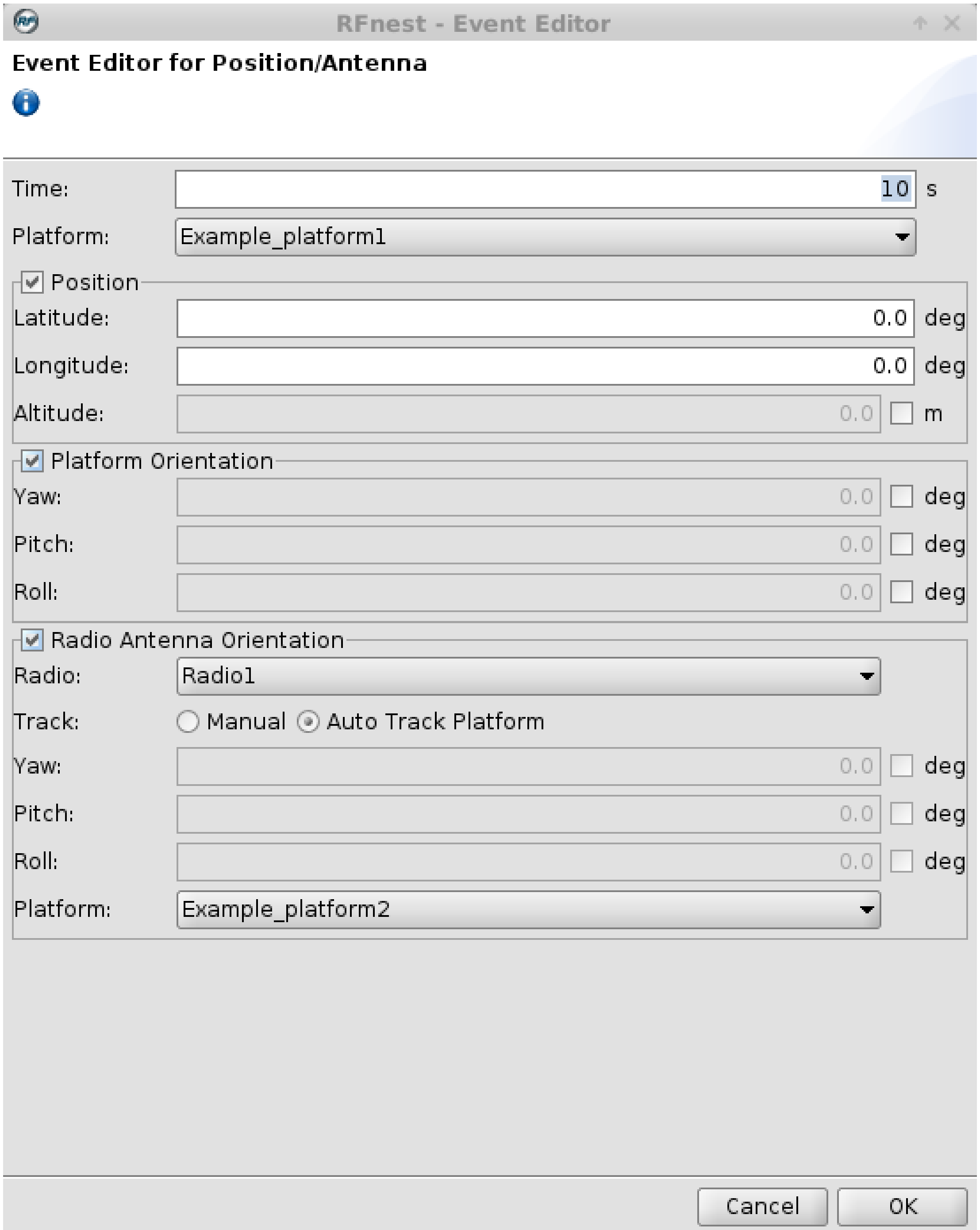
- If 'Transmit' or 'Receive' trigger condition is chosen, select the radio that would trigger record/replay from the appropriate drop-down menu.

9. Set replay delay in seconds from in the 'Replay Delay' field.
10. Set the path and filename to save the recording in the 'Filename' field.
    - Warning: "Certain RF signals may be considered sensitive information and should not be stored on the RFnest GUI platform."
11. Click 'OK' to save or 'Cancel' to cancel the event.

11.2.3 Create Position/Antenna Event

This menu enables the user to configure changes to the position and orientation of the platform and antenna orientation at a specific time in the scenario (for example if a directional antenna is going to be repositioned in an attempt to find an alternate platform to communicate with.) This event will take effect at the time specified, and the platform will remain in the position/orientation or have the antenna orientation set in the event, thus the route and waypoints will not affect the location of the platform anymore.

1. Select 'Position/Antenna Event' from the 'Event' drop-down menu and click 'New'.



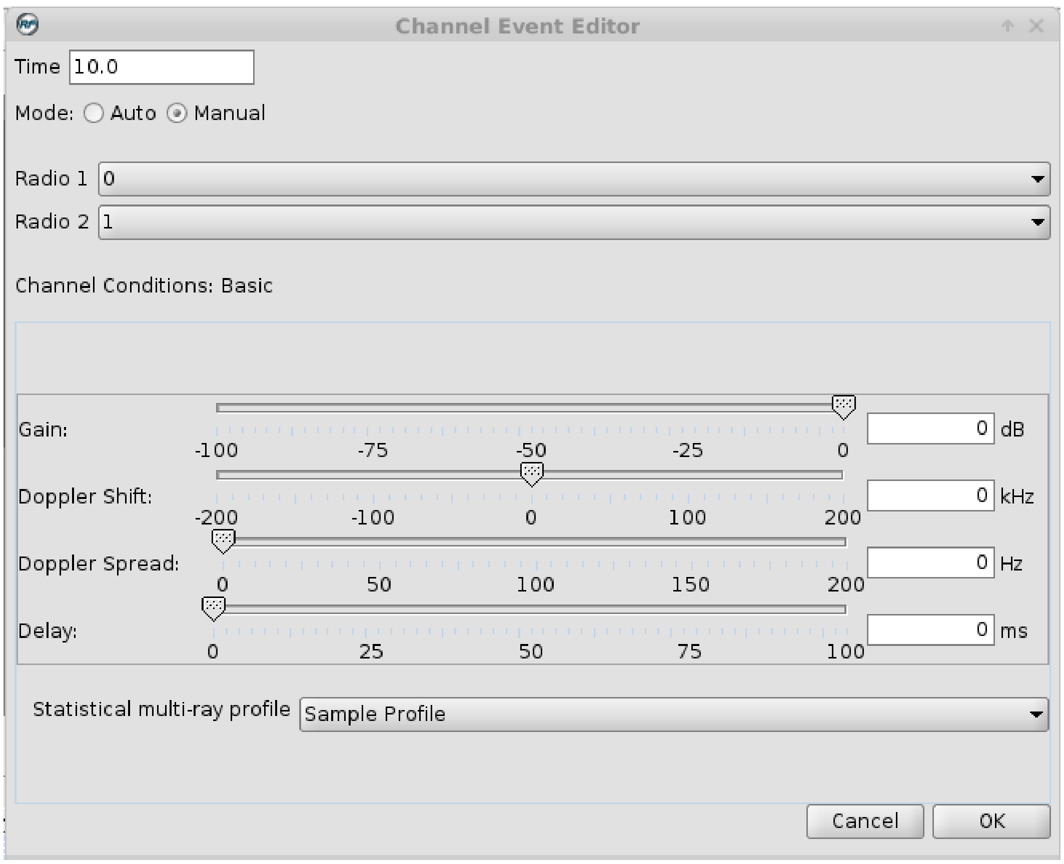
2. Set the time when of the event execution during the scenario in the 'Time' field. If 0 is set for the 'Time" field, the event will fire when the scenario starts, i.e., at 0.0 time.
3. Select the platform from the 'Platform' drop-down menu.
4. Check 'Position' checkbox to create a position event.
5. Set the latitude and longitude coordinates in the 'Latitude' and 'Longitude' fields.
6. Check the 'Altitude' filed checkbox and set altitude for position event.
7. Check the 'Platform Orientation' checkbox and the desired 'Yaw', 'Pitch', and/or 'Roll' checkbox to set platform orientation.

8. Check the 'Radio Antenna Orientation' checkbox to create/add a radio orientation event.
9. Select the radio from the previously selected platform in the 'Radio' drop-down box.
10. Select the tracking type to be 'Manual' or to 'Auto Track Platform' using the 'Track' radio buttons.
    - 'Auto Track Platform' option will make the antenna orientation follow a platform selected from the 'Platform' drop-down box.
    - 'Manual' mode will require checking 'Yaw, 'Pitch', 'Roll' checkboxes and setting the values. In this case the antenna will always be pointing in this direction, relative to the platform.
11. Click 'OK' to save or 'Cancel' to cancel the event.

11.2.4 Create Channel Event

This menu allows the user to manually specify a model between two channels at any given time (overwriting the nominal channel model). Additionally, the user has the ability to vary different aspects of the model independently to observe the network effects.

1. Select 'Channel Event' from the 'Event' drop-down menu and click 'New'.



2. Set the time when of the event execution during the scenario in the 'Time' field. If 0 is set for the 'Time" field, the event will fire when the scenario starts, i.e., at 0.0 time.
3. Select 'Auto' or 'Manual' mode from the 'Mode' radio buttons.
    - In 'Auto' mode the event will trigger at selected time but will not preserve its state if the nodes are in motion or moved by the user.
    - In 'Manual' mode the event will trigger at selected time and the channel settings will stay unless the user creates another event to set it to 'Auto' mode or sets the mode to 'Auto' in the 'Channel Properties Editor'.

4. Select the two radios whose channel will be modified by this event from the 'Radio 1' and 'Radio 2' drop-down menus.
5. Set channel gain dB by using the 'Gain' slider or by typing it in the field and pressing 'Enter' on the keyboard. Note: this field is only present if supported by the current RFnest hardware.
6. Set channel doppler shift in kHz effect by using the 'Doppler Shift' slider or by typing it in the field and pressing 'Enter' on the keyboard. Note: this field is only present if supported by the current RFnest hardware.
7. Set channel doppler spread in Hz effect by using the 'Doppler Spread' slider or by typing it in the field and pressing 'Enter' on the keyboard. Note: this field is only present if supported by the current RFnest hardware.
8. Set channel delay effect in miliseconds by using the 'Delay' slider or by typing it in the field and pressing 'Enter' on the keyboard. Note: this field is only present if supported by the current RFnest hardware.
9. Set statistical multi-ray profile by selecting it from the 'Statistical multi-ray profile' drop down menu. Note: this field is only present if supported by the current RFnest hardware.
10. Click 'OK' to save or 'Cancel' to cancel the event.

11.2.5 Edit Event

1. Open the 'Event Scheduling Wizard' by going to the 'Tools' menu and selecting 'Events' or clicking on the 'Events' icon on the toolbar.
2. Select an event from the list.
3. Click 'Edit' to bring up event editor.
4. Edit desired options.
5. Click 'OK' to save or 'Cancel' to dismiss changes.

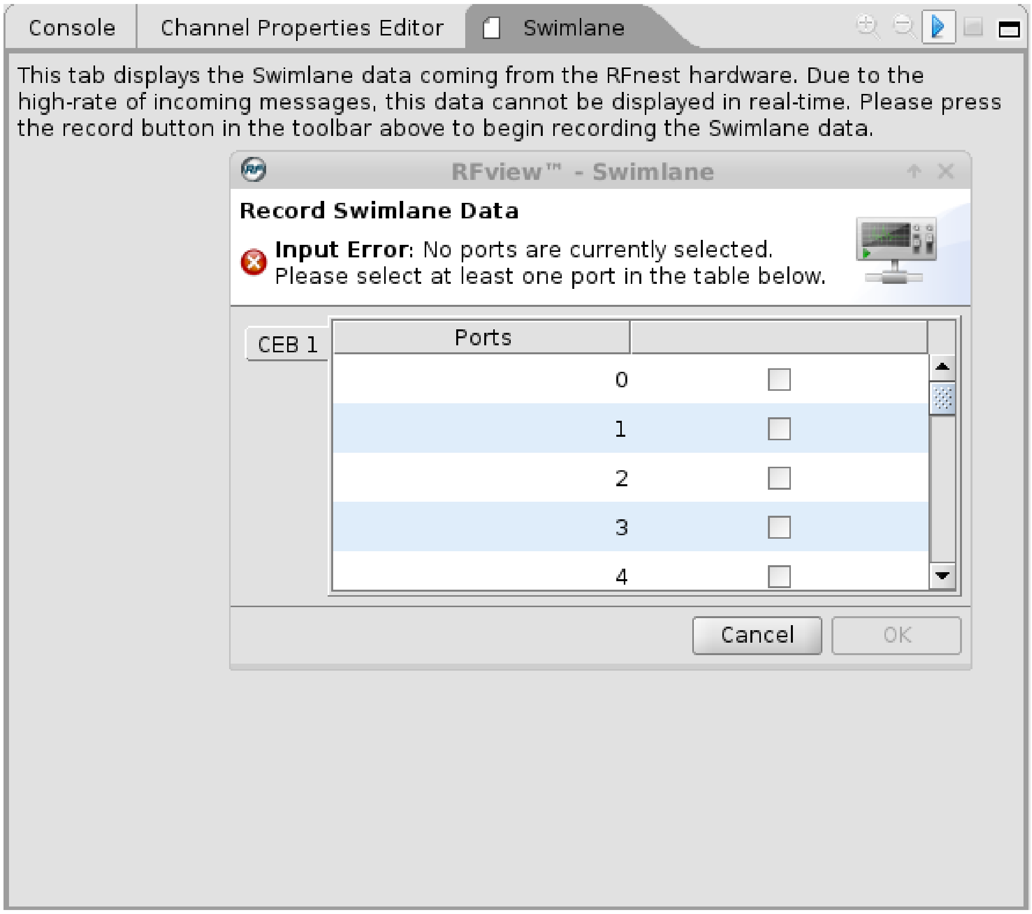
11.2.6 Delete Event

1. Open the 'Event Scheduling Wizard' by going to the 'Tools' menu and selecting 'Events' or clicking on the 'Events' icon on the toolbar.
2. Select an event from the list.
3. Click 'Delete' to remove event.

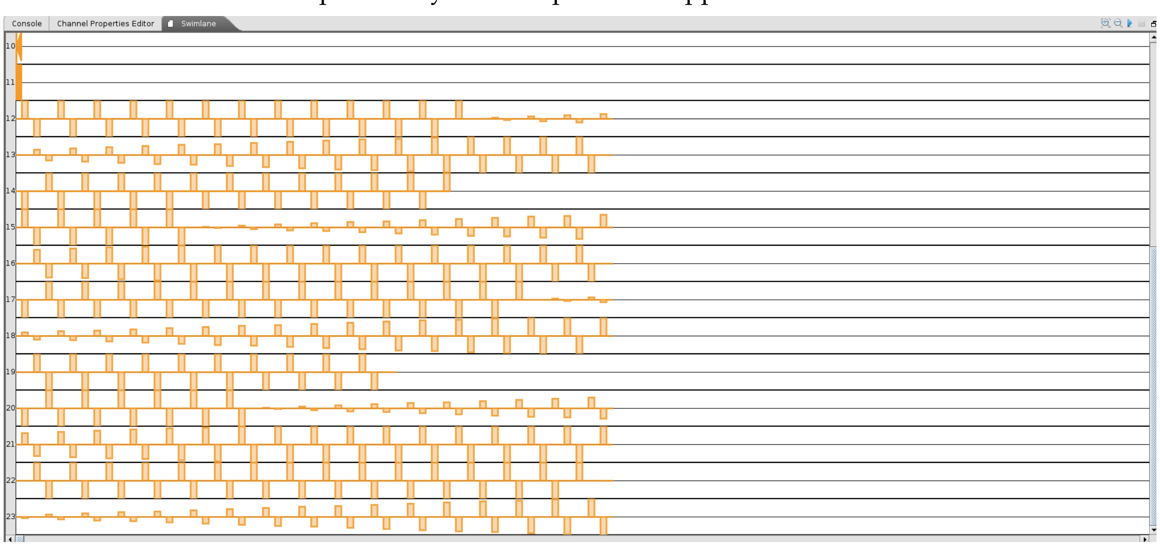
11.3 Swimlane (D-Series Only)

Swimlane tab provides the user with information about the time, duration, power, and status of the signal on hardware ports.

1. Select the 'Swimlane' tab.
2. Click on the blue play button.



3. In the 'Swimlane' options editor select desired ports to capture output.
4. Click 'OK' to start capturing or 'Cancel' to cancel capture.
5. Click on the blue stop button to stop capture.
6. The information about the previously selected ports will appear in the 'Swimlane' tab.



7. Hover over a desired point to see more information about the port, time, duration, power, and status of the signal.
8. Click on '+' or '-' icons to zoom in and out.

11.4 Automatic Gain Control (D-Series Only)

To better operate with devices which vary their transmit power and to improve the effective dynamic range of the system, an automatic gain control (AGC) function is provided. This function is **not** a replacement for properly setting the input gains at each port (Section 9.6.2) or for understanding the proper input and output levels of the system (Sections 9.5 and 9.6). Please ensure these sections have been carefully followed and that the sytem and scenario is functioning normally **prior** to attempting to use AGC.

AGC operates in the analog domain by automatically adjusting the input and output gains to optimize the signal level being digitized and counteract this adjustment so the signal is less noisy but the effective output power is unchanged. AGC is configured and performed on a per-frequency group basis. If the AGC determines that all input signals are too low, the input gain is increased and the output gain is decreased. Similarly, if any input signal is too high, the input gain is decreased and the output gain is increased. The parameters of individual channels are unchanged. **In fact, the end result of a properly configured AGC is that no AGC-related behavior is observable in the output RF signals.** It is important to understand that the user will not observe significant changes in the output power level as a result of AGC. The changes to the input gain serve only to improve the digitization of the signal, and the changes to the output gain offset any power level impacts. The net result of the AGC is a less noisy, but otherwise identical, output signal.

AGC adjusts all ports within a given frequency group simultaneously. Every second, the AGC receives measurements of the input signal level. Using the data collected during 3 to 5 seconds, the AGC compares these values against the Desired Auto Level Power (typically 0 dBm) for the frequency group and makes a decision whether to make any adjustments. If the any port's input power level exceeds the Desired Auto Level Power, the input gains of **all** ports in the frequency group are decreased. If **no** port's input power level reaches this level, the input gains of **all** ports in the frequency group are increased. Again, offsetting changes to the output gains are simultaneously applied. Essentially, the goal of the AGC is to attempt to maintain the highest input level of any port in the frequency group at the Desired Auto Level Power. When the scenario starts, the system calculates the minimum and maximum AGC values that can be used without violating the ability for the system to offset the AGC.

The AGC only operates when the scenario is running. When the scenario is started by pushing the play button, the AGC automatically examines the current gain settings and re-calculcates the Min, Max, and Current AGC values for each frequency group where AGC is enabled. The AGC then begins adjusting the gains. When the scenario is stopped, all gains revert back to the original values set prior to the scenario being played. If the scenario is paused, the AGC stops collecting or analyzing input signal levels until the scenario is resumed.

11.4.1 Using AGC

To use AGC, follow the following steps:

1. Configure the scenario as usual and use the Group Configuration dialog (Section 8.4) to add several radios to create a frequency group.

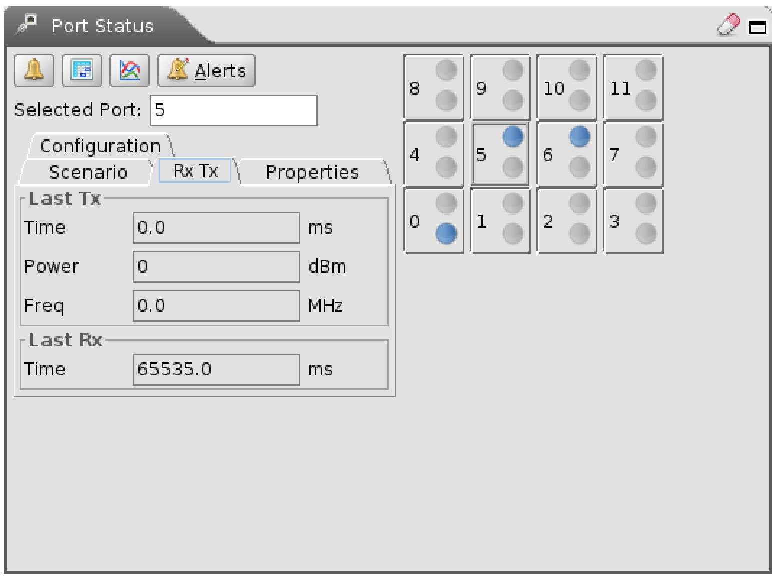
2. Run the scenario without using AGC, verify everything is correct. It is not recommended to use AGC until the normal operation of the radios in the system without AGC has been successful.

3. Initialize the system.

4. **Carefully** configure the radios to operate at the highest transmit power they will use during the scenario. Note that the ideal safe input signal level at the SMA connector is approximately -20 dBm. Prolonged input levels above -10 dBm may damage the system, especially if the input gains are set incorrectly. The AGC is not a substitute for correctly operating at a given transmit power or a way to significantly reduce the input signal level. **The input gain stage can only decrease the input level by 9 dB, while it can increase the input level by 39 dB. Thus the radio's transmit power should fluctuate such that the signal level is either at the desired input strength (i.e. -20 dBm, in which case the system will set the input gain to approximately 0 dB) or below the desired input strength**

(e.g. -40 dBm, in which case the system will apply a significant input gain). The AGC cannot successfully attenuate a signal that is significantly too strong. When a radio or other device capable of transmitting above 1 mW is attached (this includes almost any radio), attenuators rated for the radio's maximum transmit power must be used prior to the signal entering the system. If the radio has a single transmit/receive connector and a splitter is used, be sure a very strong signal is not unintentionally entering the system through the "output" RF cable.

5. With the radios transmitting continuously or at least once per second at the safe highest power described in the previous step, set the input gains as described in Section 9.6.2. Note that it is best to use the Signal Tool's auto-leveling function when AGC is going to be used, since the system will continuously invoke this functionality to adjust the gains anyway. Note that AGC will operate even if the Signal Tool's auto-leveling function fails, this failure indicates only that the observe power level fluctuated during measurement. The AGC performs additional smoothing to address this and is not affected by small fluctuations. In the image below, the port status indicates that the input signal level for the selected port is 0 dBm. This should be achieved for all ports if possible, prior to activating AGC.



6. Verify that the system is operating correctly, i.e. that the radios can communicate when the channels between them are manually set to 0 dB gain in the Channel Properties Editor, and cannot communicate when the channel is set to a high loss (e.g. -200 dB gain).

7. Stop the scenario

8. Enable AGC in the frequency group's calibration section (See Section 8.4).

9. Observe the Current, Min, and Max AGC values. These values indicate the amount of room the AGC has to compensate if the input power increases or decreases.

   - Current AGC is the highest current input gain of any port in the frequency group.
   - Max AGC is the maximum value that the Current AGC value can be increased to while maintaining correct operation. The Max AGC is affected primarily by the output attenuation setting and the system's automatic calibration adjustments. A higher (less negative) output attenuation setting will increase the Max AGC, while a lower output attenuation will decrease the Max AGC. The frequency the system is operating at also has a small impact on limiting this value.
   - Min AGC is the minimum value that the Current AGC value can be increased to while maintaining correct operation. The Min AGC is affected by the output attenuation, the variation in the input gains, and to a lesser extent the system calibration for the operating frequency. The output attenuation setting has the same impact on Min AGC as on Max AGC. A higher output attenuation setting increases the Min AGC, while a lower setting decreases the Min AGC.

10. If an error or warning icon is shown in the dialog, the AGC settings are not suitable for the AGC to operate correctly. Generally this occurs because the Max is lower than the Min.

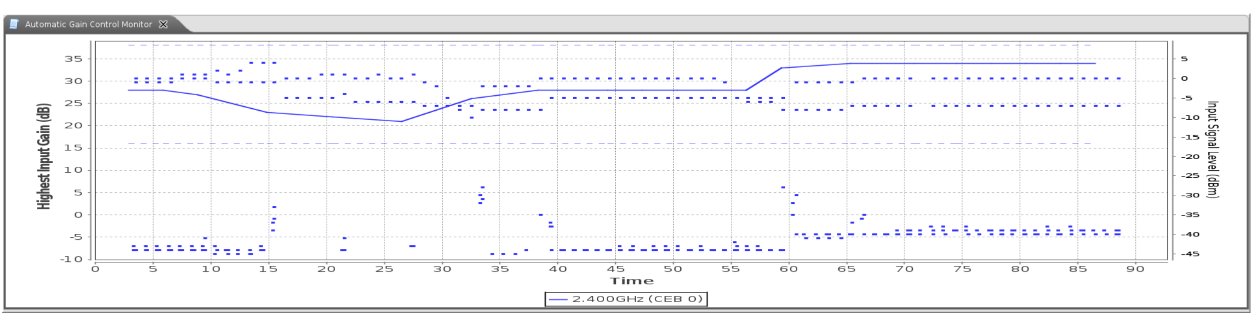
11. If the Current, Max, or Min AGC values are unsatisfactory, take steps to change them.

    - Generally the output attenuation slider should be set to -10 dB when AGC is in use. This change will allow the Max AGC value to increase and usually resolve most problems. This setting provides the system with some room to handle slightly higher input signals than are currently in use, and typically 15-20 dB of room to handle weaker input signals. If more output attenuation is desired, the user should add fixed attenuators external to the system on the "output" RF cables connected to the radios. At -10 dB output attenuation, a Max AGC of 38 dB (the maximum achievable value) is generally attained.
    - To decrease the Min AGC, setting the output attenuation to -10 dB is the first step as well. If a lower value is still needed, change the input gains so they are closer together. To do this without negatively affecting performance, add or remove attenuators to the "input" RF cable attached to the radios to adjust the levels. If the radio attached to a port is acting only as a receiver, set that port's input gain to the same as the transmitters' input gains. The Min AGC is constrained by the fact that all input gains are increased/decreased in unison. Thus whichever port has the lowest input gain at the start of the scenario is the limiting factor preventing a lower Min AGC from being possible, since that gain can be decreased by the smallest amount before hitting the lowest possible input gain setting.
    - To change the Current AGC (i.e. to move it away from the Min or Max to create more room), simply adjust the input gain of the port in this frequency group with the highest input gain.
    - Note that after adjusting the input gains via the Port Status widget or auto-leveling, re-entering the Group Configuration dialog will re-calculate all of the AGC values for display to the user. The AGC will automatically recalculate these values for its own use even if this is dialog is not opened.

12. Re-initialize the system. This will apply any settings changed in the Group Configuration dialog. The system can now be run with AGC.

11.4.2 AGC Monitor

Under the Tools, View menu, select the Automatic Gain Control Monitor. This will create a graph view showing the state of the AGC. This view is only enabled when at least one frequency group has AGC enabled and the scenario is running. Each frequency group is shown by a different color in the graph. The Min and Max AGC values are shown as a thin dashed line. The Current AGC value is shown as a thick solid line. These values are plotted using the left hand Y axis scale. As the AGC adjusts up or down, these lines show the AGC's current position relative to the minimum and maximum. Ideally, the solid line (Current AGC value) is never at either dashed line (the Min and Max AGC value), but is instead always somewhere between the two. These Current AGC is updated whenever it is recalculated, every 3 to 5 seconds. The Min and Max AGC are always constant for the duration of the scenario.



On the same graph, the input signal levels are shown for all ports in the frequency group as a small dot using the right hand side Y axis scale. This is the same value displayed in the Port Status widget for the ports. If the highest input signal level deviates from the Desired Auto Level Power (typically 0 dBm), the

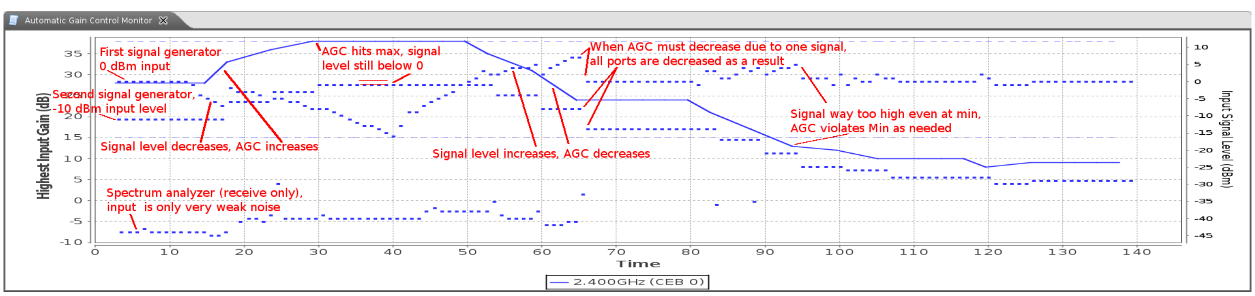
AGC will adjust accordingly. Ideally, the highest value among all input signal levels reported is always within 1-2 dB of the Desired Auto Level Power. The other values may be lower. For ports with a lower input signal level, the digitization of the RF signal will be less optimal. Since the input signal levels are on a separate Y axis scale, they should not be expected to coincide with the absolute values of the AGC lines with the exception that a relative change in one will trigger a corresponding relative change in the other.

By looking at this graph, it can be observed that the AGC decreases when the input signal level exceeds 0 dBm (The Desired Auto Level Power in this example), and increases when all input signal levels are below 0 dBm. Note that the input signal level measurements at approximately -40 dBm are those of a spectrum analyzer which was acting as a receiver only. A signal level at or below -40 dBm typically indicates that only ambient noise, and not an actual signal, is being received on a given port.

The input signal levels can be used by the user to determine what the behavior of the radios is and identify whether some or all ports have input signal levels that are too high or too low. To determine which ports are reporting a given value, the Port Status widget can be used to view the radio's Tx power levels under the "Rx Tx" tab by selecting each port one by one.

11.4.3 AGC Behavior Example

To illustrate the AGC behavior, consider the example below. In this case, 2 signal generators and 1 spectrum analyzer are all attached to RFnest, forming a frequency group with 3 "radios." AGC is enabled for this frequency group and the Desired Auto Level Power is set to 0 dBm. At the start of the scenario, the first signal generator is transmitting at -20 dBm and the input gain has been adjusted so that port achieves 0 dBm input signal level. The second signal generator is also set to transmit at -20 dBm. This port is intentionally incorrectly adjusted to achieve only -10 dBm to illustrate the AGC behavior when multiple devices with different input signal levels are present. In normal usage, the input gain should be adjusted such that a 0 dBm input signal level is achieved for all ports attached to devices with a fixed transmit power. Finally, a spectrum analyzer is attached to the third port in the frequency group. Since the spectrum analyzer does not transmit any signal, the system reports a very low input signal level from this port which is a measurement of only the noise floor in the RF front end with an unterminated "input" RF connection. The input gain for this port is set to match that of the second signal generator's port, to prevent a low input gain from unnecessarily restricting the AGC's minimum value.



Initially, when the scenario starts the first signal generator's input level is already at 0 dBm and the first signal generator's transmit power is unchanged. Thus the AGC remains constant. Note that it is only a coincidence that the highest input signal level (at 0 dBm according to the right hand Y axis) and the AGC value (28 dB according to the left hand Y axis) appear at the same level in the graph for the first 10 seconds. It is important to observe that for the first 10 seconds, the AGC does not and can not act in a way that would increase the input signal level of the second signal generator or the very low noise measured from the signal generator's port. This is because the AGC moves all input signal levels up/down in unison to preserve channel correctness. Increasing the AGC to improve these input signal levels would raise the first signal generator's input signal level above 0 dBm. At approximately time 14, the first signal generator's transmit power is decreased. The AGC detects this after approximately time 17 and increases the gains of all 3 ports by approximately 5 dB. This causes the input signal levels to jump up by approximately 5 dB. At this point, the input signal level of the first signal generator is approximately 0 dBm and the second signal generator's input signal level has increased to approximately -5 dBm. This increase in the second signal generator's input signal level is an indirect result of the AGC attempting to keep the highest input signal

level of all ports at 0 dBm, which caused the input signal levels of **all** ports to be increased. The direct cause for the two input signal levels becoming closer together is that the first signal generator's input signal level became lower (closer) to the second's, not because of the AGC.

From time 17 to time 30, the AGC continues to increase as the first signal generator's transmit power is decreased further. It is somewhat counterintuitive that the input signal level continues to be low since the AGC should be moving to offset it. However, the transmit power is being decreased so rapidly that although the AGC has moved so that the input signal level at the time the decision was made has become zero, by the time the decision is made the signal has decreased further. This is typically not an issue, as sub-optimally digitizing the signal a few dB below 0 dBm has minimal impact on performance. Around time 25, the second signal generator's input signal level becomes the highest (which dots correspond to which ports' input signal levels can only be inferred from the graph, or can be observed directly by examining each port's current value in the Port Status widget). The AGC now switches to attempting to make the input signal level of the second signal generator, rather than the first's, at 0 dBm. At time 30, the AGC hits the maximum value. If the AGC increases further, there is no room in the output gains to decrease further, meaning the input gain would not be fully offset by the output gains. The AGC settles for sub-optimal digitization of the signals rather than incorrect output signal levels. Due to this restriction, during this time no input signal levels are at 0 dBm. Note also that the input signal level of the first signal generator continues to decrease as the first signal generator continues to decrease its transmit power.

At time 50, the first signal generator's transmit power increases such that the input signal level exceeds 0 dBm and exceeds that of the second signal generator. The AGC reacts to this by adjusting down to keep all input signals at or below 0 dBm. As the first signal generator's transmit power continues to be increased, the AGC continues to decrease until at time 65 when the first signal generator's transmit power stays fixed at the current level. From times 65 to time 80, the AGC maintains the first signal generator's input signal level at 0 dBm. Observe that during this time period the relative difference between the input signal levels of the two signal generators is higher than it was originally from time 0 to 10. Again, the AGC cannot take any action to improve the relative different in signal levels, the relative difference is dictated by the user's initial input gain settings and by the relative changes in the transmit power during the scenario.

From time 80 to 90, the first signal generator's input signal level increases further to the point that even at the minimum correct value, the AGC cannot maintain the signal below 3 dBm (the safety threshold, and also a point where noise due to saturation is occurring). The AGC therefore moves below the minimum value and continues to stay below even when the input signal level becomes zero, since a move back to the minimum would cause the input signal level to exceed 3 dBm.

11.4.4 AGC Alerts and Logging

The AGC behavior is logged and has the ability to alert the user when the AGC goes out of bounds. At each decision point, an Automatic Gain Control Event 11.7.2.7 is logged to the event deck containing the Current, Min, and Max AGC values for each frequency group where AGC is enabled. This data can be used combined with Signal Status Update Events 11.7.2.8 to analyze system behavior in more detail.

If configured, the system generates an Automatic Gain Control Out of Range alert for each offending port causing the AGC to be within a tolerance of the Min and Max AGC. See Section 9.3.4 for details.

11.4.5 AGC Caveats

The following caveats may be encountered in certain situations when using the AGC.

1. **The AGC does not fix the situation where too strong or too weak of an input signal level is input into the system.** The input signal strength per the specifications should be between -20 dBm and -40 dBm. The AGC can adjust signals at or near this strength, but is designed to handle radios that change their transmit power dynamically within this range, not that operate outside this range. If the signal levels are too high or too low, external attenuation must be added or removed.
2. **The AGC does not protect the system from strong input levels.** If a very strong signal level is input

to the system (i.e. 0 dBm or higher), the AGC can only reduce the signal level by 10 dB. In addition to the signal being not optimally digitized, there is a risk of thermal damage to the hardware.

3. The AGC assumes the user has set the initial input gains correctly for all ports. If no port is set such that the input signal level is at or near the Desired Auto Level Power (typically 0 dBm), the AGC will trigger a sudden jump in the input gains at the start of the scenario. While 0 dBm is strongly preferred since it indicated optimal digitization of the RF signal, if 0 dBm is not achievable for some reason, decrease the Desired Auto Level Power to prevent AGC misbehavior.

4. If some ports generate an input signal near the Desired Auto Level Power but others generate a much lower signal, the AGC cannot improve this situation. In cases where the transmit power of the radios on these ports is fixed this is a configuration error, i.e. the input gain or external attenuation should be adjusted prior to the start of the scenario such that all ports achieve an input signal level close to the Desired Auto Level Power. In other cases, one radio may change its transmit power to be higher while another may change its power to be lower. In this case, the AGC must act on the higher transmit power, causing the more weakly transmitting radio's RF to be sub-optimally digitized. Essentially, the AGC will never act in a way that will make the input signal levels of the ports within a frequency group closer to each other. The AGC only adjusts all levels up and down, and moves the levels of all ports simultaneously up/down by the same amount. Otherwise, the AGC would violate the integrity of the channels configured in the scenario.

5. The AGC re-evaluates the gains approximately every 3 to 5 seconds. This delay means that a radio which quickly changes transmit power by a significant amount will briefly have an input signal level that is too high or too low until the next decision point occurs. If the radios change their transmit power by only a few dB at a time, this will not pose a problem. For radios which change their transmit powers by 5, 10, or more dB at a very fast rate some transmissions may always be sub-optimally digitized.

6. The AGC assumes each transmitting radio transmits at least once every 3 seconds. If the radio with the highest current input signal level does not transmit for more than 3 seconds, the next highest input signal level will be used to control the AGC until the radio begins transmitting again. This may cause the intermittently transmitting radio's signal to be temporarily too high when it first resumes transmitting until the next decision point occurs and the AGC begins considering it again.

7. The AGC will not violate the Max AGC value. If the input signal levels are still too weak and the Max AGC is reached, the signals will be digitized sub-optimally.

8. The AGC **will** violate the Min AGC value. If the input signal levels are too high, the AGC will not hesitate to decrease the input signal levels as low as necessary to prevent the input signal from being too high. The Min AGC is shown as a reference only, and is not a constraint. If the AGC goes below this value, the output signals for some or all ports will be somewhat weaker than they should be because the output gains cannot be increased enough to compensate for the low input gains. The AGC will remain below the maximum value until the system calculates that returning to the valid range will not cause the input signal levels to significantly exceed 2 dBm. Note that the ability of the AGC to violate the Min AGC value is not intended to be used as a safety mechanism where the user causes signal levels to be too high, and is only to prevent AGC-induced high signal levels.

9. The AGC alerts (or lack thereof) refer to the AGC state, not the input signal levels of all ports. In other words, the alerts are generated when the AGC is at or near the Min and Max AGC. The alerts are not generated when the input signal levels of some ports are satisfactory and other ports are far too low. It is up to the user to monitor the AGC behavior, to ensure that the input gains are set correctly at the start of the scenario, and to understand the implications of using the AGC.

10. When the AGC is too low (i.e. a port has too high of a transmit power), the generated alert will indicate the offending ports(s).

11. When the AGC is too high (i.e. all ports' transmit powers are too low), the alert will be generated for each port in the frequency group. The system cannot determine which ports should/shouldn't be transmitting, etc. and so flags all of them with an alert status.

11.5 Antenna Pattern Files

To set antenna pattern for a node refer to section 2 on how to create/edit a radio. Antenna type 1(Antenna1.pat) is the default omnidirectional antenna, other patterns apply to directional antennas and should have a corresponding antenna pattern file in the cec/antenna/ folder. These patter files should follow EMANE antenna pattern file format, and have to be named antenna.xml, where 'X' is the number(starting with 2, i.e. antenna2.xml) corresponding to the desired pattern number selected in RFview.

Below is an example of the EMANE antenna pattern file format. For more information please refer to the EMANE manual at http://downloads.pf.itd.nrl.navy.mil/docs/emane/emane.pdf

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE antennaprofile SYSTEM
  "file:///usr/share/emane/dtd/universalphyantennaprofile.dtd">
<antennaprofile>
  <antennapattern>
    <elevation min='-90' max='-16'>
      <bearing min='0' max='359'>
        <gain value='-200'/>
      </bearing>
    </elevation>
    <elevation min='-15' max='-11'>
      <bearing min='0' max='5'>
        <gain value='0'/>
      </bearing>
      <bearing min='6' max='10'>
        <gain value='-3'/>
      </bearing>
      <bearing min='11' max='15'>
        <gain value='-6'/>
      </bearing>
      <bearing min='16' max='344'>
        <gain value='-200'/>
      </bearing>
      <bearing min='345' max='349'>
        <gain value='-6'/>
      </bearing>
      <bearing min='350' max='354'>
        <gain value='-3'/>
      </bearing>
      <bearing min='355' max='359'>
        <gain value='0'/>
      </bearing>
    </elevation>
    <elevation min='-10' max='10'>
      <bearing min='0' max='5'>
        <gain value='3'/>
      </bearing>
      <bearing min='6' max='10'>
        <gain value='0'/>
      </bearing>
      <bearing min='11' max='15'>
        <gain value='-3'/>
      </bearing>
      <bearing min='16' max='344'>
        <gain value='-200'/>
      </bearing>
      <bearing min='345' max='349'>
        <gain value='-3'/>
      </bearing>
      <bearing min='350' max='354'>
        <gain value='0'/>
      </bearing>
      <bearing min='355' max='359'>
        <gain value='3'/>
```

```
      </bearing>
    </elevation>
    <elevation min='11' max='90'>
      <bearing min='0' max='359'>
        <gain value='-200'/>
      </bearing>
    </elevation>
  </antennapattern>
</antennaprofile>
```

11.6 Virtual node configuration.

To set up virtual nodes in CORE/EMANE please follow the tutorials provided at:

CORE: http://downloads.pf.itd.nrl.navy.mil/docs/core/core-html/index.html

EMANE: http://downloads.pf.itd.nrl.navy.mil/docs/emane/emane.pdf

*Note that CEC will process all messages sent by EMANE, thus EMANE scenarios should be populated using nodes with numbers that correspond to the virtual nodes set in RFview GUI. (Ex. if real nodes in scenario are nodes 0 through 23, than EMANE simulated nodes should be number 24, 25, etc.)

To configure processing of information from CORE/EMANE virtual nodes please set the following cec.xml parameters as needed:

```
<param name="publishpathlossevents" value="on"/>
<param name="publishlocationevents" value="on"/>
<param name="processpathlossevents" value="on"/>
<param name="processlocationevents" value="on"/>
```

publishpathlossevents With this parameter set to 'on' cec will send pathloss values for all channels to EMANE.

publishlocationevents With this parameter set to 'on' cec will send location information for all nodes to EMANE.

processpathlossevents With this parameter set to 'on' cec will be able to receive and apply pathloss values for channels sent from EMANE.

processlocationevents With this parameter set to 'on' cec will be able to receive location information sent from EMANE and calculate pathloss based on EMANE locations.

11.7 Event Deck

EventDeckExporter is a command-line utility provided by RFview that converts the contents of the event logs and prints them to the standard out. This is a useful tool for exporting data for other tools (e.g. to graph data), detailed manual analysis, and for debugging scenarios that contain a large number Platforms and Radios.

11.7.1 EventDeckExporter Usage

To start the EventDeckExporter, execute the companion script rfview-eventexporter. This application takes one command-line argument pointing to the directory containing the recorded events. The application is specifically looking for two files within this directory `event_deck.idx` and `event_deck.records` which contain the event index of the recording and the events of the recording respectively.

The RFview workbench automatically maintains a registry of scenario recordings in the scenario's output directory. The name of the directory is based on the name of the scenario. For example if the scenario is named `testScenario` the scenario's XML file name will be `testScenario.xml` and the scenario's output directory is `testScenario_output/`. Under the output directory, there will be a subdirectory for each scenario recording starting with 000.

Linux users should use the shell script `rfnest-eventexporter.sh`:

```
./rfnest-eventexporter.sh /path/to/scenario/testScenario_output/000
```

Windows users should use the batch script `rfnest-eventexporter.bat`:

```
rfnest-eventexporter.bat /path/to/scenario/testScenario_output/000
```

The EventDeckExporter can also be launching using a manual classpath using the following command:

```
java -cp /path/to/rfview.jar com.iai.rfnest.core.util.EventDeckExporter \
  /path/to/scenario/testScenario_output/000
```

11.7.2 Event Formats

Each line outputted by the EventDeckExporter starts with the scenario time, in seconds. All events have their own custom formatting. For specific details about the contents of each event, see the section below.

11.7.2.1 Position Update Event

Position Updates are logged to the event deck each time the position of a platform changes within the scenario. For each occurrence on the Event Deck, a position update message will also be transmitted to the CEC.

Content:

Name	Description	
Type	Always PositionUpdateEvent	Required
PlatformId	Contains the internal identifier of the Platform.	Required
PlatformName	Contains the name of the Platform. This property is not located within the contents of the event, but the EventDeckExporter will attempt to determine the name of the platform based on the scenario file. If it cannot be determined, Unknown will be used.	Optional
Latitude	Contains the Latitude Component of the Platform's location.	Required
Longitude	Contains the Longitude Component of the Platform's location.	Required
Altitude	Contains the Altitude Component of the Platform's location.	Required

Example Output:

```
[000046.800] Type: PositionUpdateEvent PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1 Latitude: 36.526689764434366 Longitude: -78.29770027512058
 Altitude: 2869.6838362611684
[000046.900] Type: PositionUpdateEvent PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1 Latitude: 36.5267315051491 Longitude: -78.29776604748575 Altitude:
 2869.405382919846
```

11.7.2.2 RSSI Event

RSSI Events are logged to the event deck each time an incoming RSSI message is received by the RFview workbench.

Contents:

Name	Description	
Type	Always RssiEvent	Required
SourceRadio	Contains the port number of the source radio of the link.	Required
DestinationRadio	Contains the port number of the destination radio of the link.	Required
SignalStrength	Contains the signal strength value of the given link.	Required

Example Output:

```
[000046.198] Type: RssiEvent SourceRadio: 4 DestinationRadio: 5 SignalStrength: -110
[000046.198] Type: RssiEvent SourceRadio: 6 DestinationRadio: 7 SignalStrength: -110
[000046.199] Type: RssiEvent SourceRadio: 1 DestinationRadio: 2 SignalStrength: -10
[000046.200] Type: RssiEvent SourceRadio: 3 DestinationRadio: 4 SignalStrength: -50
[000046.200] Type: RssiEvent SourceRadio: 5 DestinationRadio: 6 SignalStrength: -100
```

11.7.2.3 Packet Route Event

Packet Route Events are logged to the event deck each time an incoming Packet Route message is received by the RFview workbench.

Contents:

Name	Description	
Type	Always PacketRouteEvent	Required
PacketRouteId	Contains the identifier of the packet route. This is an internally property of the RFview workbench used for book keeping.	Required
Protocol	Contains the protocol for the given link.	Required
SourceRadio	Contains the port number of the source radio of the link.	Required

Example Output:

```
[046.213] Type: PacketRouteEvent PacketRouteId: 316 Protocol: 0 SourceRadio: 1 DestinationRadio↩
    : 2
[046.215] Type: PacketRouteEvent PacketRouteId: 317 Protocol: 1 SourceRadio: 2 DestinationRadio↩
    : 1
[046.215] Type: PacketRouteEvent PacketRouteId: 318 Protocol: 2 SourceRadio: 3 DestinationRadio↩
    : 4
[046.216] Type: PacketRouteEvent PacketRouteId: 319 Protocol: 3 SourceRadio: 4 DestinationRadio↩
    : 3
[046.216] Type: PacketRouteEvent PacketRouteId: 320 Protocol: 4 SourceRadio: 5 DestinationRadio↩
    : 6
[046.217] Type: PacketRouteEvent PacketRouteId: 321 Protocol: 5 SourceRadio: 6 DestinationRadio↩
    : 5
```

11.7.2.4 Remove Packet Route Event

Remove Packet Route Events are logged to the event deck each time the user deletes a packet route during execution of the scenario. This is an internal event used for book keeping.

Contents:

Name	Description	
Type	Always RemovePacketRouteEvent	Required
PacketRouteId	Contains the identifier of the packet route. This is an internally property of the RFview workbench used for book keeping.	Required

Example Output:

```
[000046.208] Type: RemovePacketRouteEvent PacketRouteId: 308
[000046.209] Type: RemovePacketRouteEvent PacketRouteId: 309
[000046.209] Type: RemovePacketRouteEvent PacketRouteId: 310
[000046.210] Type: RemovePacketRouteEvent PacketRouteId: 311
[000046.211] Type: RemovePacketRouteEvent PacketRouteId: 312
[000046.211] Type: RemovePacketRouteEvent PacketRouteId: 314
[000046.211] Type: RemovePacketRouteEvent PacketRouteId: 313
[000046.213] Type: RemovePacketRouteEvent PacketRouteId: 315
```

11.7.2.5 DCU Notification Type A Event

DCU Notification Type A events are logged to the event deck each time the CEC issues a channel updated for a particular link.

Contents:

Name	Description	
Type	Always DCUANotificationEvent	Required
SourceRadio	Contains the port number of the source radio of the link.	Required
DestinationRadio	Contains the port number of the destination radio of the link.	Required
AverageLoss	Contains the average loss of the link.	Required
DopplerShift	Contains the doppler shift of the link.	Required
DopplerSpread	Contains the doppler spread of the link.	Required
Delay	Contains the delay of the link.	Required
ControlMode	Contains the control mode of the given link. 0 indicates that the CEC is maintaining control of the link, and 1 indicates that the operator is maintaining control of the link.	Required

Example Output:

```
[000046.750] Type: DCUANotificationEvent SourceRadio: 6 DestinationRadio: 7 AverageLoss:
 -117.11293753524603 DopplerShift: 128 DopplerSpread: -12281 Delay: 100000 ControlMode: 0
[000046.765] Type: DCUANotificationEvent SourceRadio: 0 DestinationRadio: 7 AverageLoss:
 -124.64969294572688 DopplerShift: 128 DopplerSpread: -12281 Delay: 100000 ControlMode: 0
[000046.780] Type: DCUANotificationEvent SourceRadio: 1 DestinationRadio: 7 AverageLoss:
 -123.93183216558154 DopplerShift: 128 DopplerSpread: -12281 Delay: 100000 ControlMode: 0
```

11.7.2.6 Perf Data Event

Perf Data, or Performance Data, events are logged to the event deck each time the RFview workbench receives a graph request.

Contents:

Name	Description	
Type	Always PerfDataEvent	Required
Port	Contains the local UDP port the data point was received on. The workbench uses this field to determine which graph to update.	Required
DataPoint	Contains the data point.	Required

Example Output:

```
[000046.740] Type: PerfDataEvent Port: 20862 DataPoint: 875.0
[000046.740] Type: PerfDataEvent Port: 20863 DataPoint: 975.0
[000046.792] Type: PerfDataEvent Port: 20862 DataPoint: 876.0
[000046.792] Type: PerfDataEvent Port: 20863 DataPoint: 976.0
```

11.7.2.7 Automatic Gain Control Event (D-Series Only)

An Automatic Gain Control Event is logged to the event deck each time an AGC decision point is reached, even if the AGC remains the same after the decision.

Contents:

Name	Description	
Type	Always AutomaticGainControlEvent	Required
Ceb Id	Contains the identifier of the CEB for the event.	Required
Frequency	Contains the frequency in MHz of the frequency group.	Required
Adjustment	Indicates the current AGC value.	Required
Min	The minimum value the AGC may choose without impacting correctness.	Required
Max	The maximum value the AGC may choose without impacting correctness.	Required

Example Output:

```
[0007.0] Type: AutomaticGainControlEvent Ceb Id: 0 Frequency: 2400 Adjustment: 17 Min: 13 Max: ↩
    38
[0010.0] Type: AutomaticGainControlEvent Ceb Id: 0 Frequency: 2400 Adjustment: 16 Min: 13 Max: ↩
    38
```

11.7.2.8 Signal Status Update Event (D-Series Only)

4 signal status update events are logged to the event deck for each signal status update message received from the hardware.

Contents:

Name	Description	
Type	Always SSUEvent	Required
Ceb	Contains the identifier of the CEB for the event.	Required
Ddb	Contains the identifier of the DDB of the CEB for the event.	Required
Port	Indicates the port of the CEB for the event.	Required
LastTx	The last transmit time.	Required
LastRx	The last receive time.	Required
SignalStrength	The raw signal strength reported by the hardware (add 26 to get the value shown in RFview).	Required
Frequency	The frequency.	Required

Example Output:

```
[000001.000] Type: SSUEvent Ceb: 0 Ddb: 0 Port: 0 LastTx: 0 LastRx: 0 SignalStrength: -20
 Frequency: 2400
[000001.000] Type: SSUEvent Ceb: 0 Ddb: 0 Port: 1 LastTx: 100 LastRx: 499 SignalStrength: -19
 Frequency: 2401
[000001.000] Type: SSUEvent Ceb: 0 Ddb: 0 Port: 2 LastTx: 200 LastRx: 998 SignalStrength: -18
 Frequency: 2402
[000001.000] Type: SSUEvent Ceb: 0 Ddb: 0 Port: 3 LastTx: 300 LastRx: 1497 SignalStrength: -17
 Frequency: 2403
[000002.000] Type: SSUEvent Ceb: 0 Ddb: 1 Port: 4 LastTx: 0 LastRx: 0 SignalStrength: -20
 Frequency: 2400
[000002.000] Type: SSUEvent Ceb: 0 Ddb: 1 Port: 5 LastTx: 100 LastRx: 499 SignalStrength:
 -19 Frequency: 2401
[000002.000] Type: SSUEvent Ceb: 0 Ddb: 1 Port: 6 LastTx: 200 LastRx: 998 SignalStrength:
 -18 Frequency: 2402
[000002.000] Type: SSUEvent Ceb: 0 Ddb: 1 Port: 7 LastTx: 300 LastRx: 1497 SignalStrength:
 -17 Frequency: 2403
```

11.7.2.9 Swimlane Event (D-Series Only)

Swimlane events are logged to the event deck for each Swimlane sample point received by the RFview workbench. A single Swimlane message received by the workbench consists of up to 1400 sample points.

Contents:

Name	Description	
Type	Always SwimlaneEvent	Required
Ceb	Contains the identifier of the CEB for the sample point.	Required
Port	Indicates the port of the CEB the sample point corresponds to.	Required
SignalPower	Contains the signal power.	Required
Status	Indicates that status (Idle: 0, Rx: 1, or Tx: 2).	Required
Duration	Contains the duration of the event in nanoseconds.	Required

Example Output:

```
[2198287.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 1280.0
[2199567.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -77 Status: 1 Duration: 128.0
[2199695.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 1792.0
```

```
[2201487.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 2048.0
[2203535.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 640.0
[2204175.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 1152.0
[2205327.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 1024.0
[2206351.000] Type: SwimlaneEvent Ceb: 0 Port: 23 SignalPower: -78 Status: 1 Duration: 2048.0
```

11.7.2.10 Set Radio Parameters Event

Set Radio Parameters is a request issued by RFview during initialization. It contains the raw contents of a message to be transmitted to the CEC.

Contents:

Name	Description	
Type	Always SetRadioParametersEvent	Required
Message	Contains the raw bytes of the message. Byte values are displayed in hexadecimal in the following format: 0 -> $00, 10 -> $0A, 255 -> $FF.	Required

Example Output:

```
[2205327.000] Type: SetRadioParametersEvent Message: \$01 \$02 \$03 \$04 \$FF
[2206351.000] Type: SetRadioParametersEvent Message: \$01 \$02 \$03 \$04 \$FF
```

11.7.2.11 Manual Channel Event

Manual Channel events are logged to the event deck each time the operator modifies the channel properties for a link in the RFview workbench. The workbench will transmit a Dcu Type A message to the CEC. If the control mode of the channel state is set to manual, the CEC will no longer issue DcuNotificationTypeA messages. This event is also generated for all Channel Properties related script events.

Contents:

Name	Description	
Type	Always ManualChannelEvent	Required
SourceRadio	Contains the port number of the source radio of the link.	Required
DestinationRadio	Contains the port number of the destination radio of the link.	Required
AverageLoss	Contains the average loss of the link.	Required
DopplerShift	Contains the doppler shift of the link.	Required
DopplerSpread	Contains the doppler spread of the link.	Required
Delay	Contains the delay of the link.	Required
ControlMode	Contains the control mode of the given link. 0 indicates that the CEC is maintaining control of the link, and 1 indicates that the operator is maintaining control of the link.	Required

Example Output:

```
[000001.000] Type: ManualChannelEvent SourceRadio: 1 DestinationRadio: 2 AverageLoss: 0.0
 DopplerShift: 0 DopplerSpread: 0 Delay: 0 ControlMode: 1
```

11.7.2.12 GeometryChangeRequest

Geometry Change Request events are generated for Position/Antenna related script events.

Contents:

Name	Description	
Type	Always GeometryChangeRequest	Required
PlatformId	Indicates the identifier of the platform for platform position and platform orientation events.	Optional
PlatformName	Indicates the name of the platform for platform position and platform orientation events.	Optional
RadioId	Indicates the identifier of the platform for antenna orientation events.	Optional
RadioName	Indicates the nam eof the platform for antenna orientation events.	Optional
PlatformLatitude	Indicates the latitude component of the desired position for the platform.	Optional
PlatformLongitude	Indicates the longitude component of the desired position for the platform.	Optional
PlatformAltitude	Indicates the altitude component of the desired position for the platform.	Optional
PlatformRoll	Indicates the roll component of the desired orientation for the platform.	Optional
PlatformPitch	Indicates the pitch component of the desired orientation for the platform.	Optional
PlatformYaw	Indicates the yaw component of the desired orientation for the platform.	Optional
RadioRoll	Indicates the roll component of the desired orientation for the radio.	Optional
RadioPitch	Indicates the pitch component of the desired orientation for the radio.	Optional
RadioYaw	Indicates the yaw component of the desired orientation for the radio.	Optional
AutoTrackPlatformName	If auto-track is enabled, it indicates the name of the platform that the event is auto-tracking to.	Optional

Example Output:

```
[000001.000] Type: GeometryChangeRequest PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1 PlatformLatitude: 1.0 PlatformLongitude: 1.0
[000002.000] Type: GeometryChangeRequest PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1 PlatformLatitude: 1.0 PlatformLongitude: 1.0 PlatformAltitude: 100.0
[000003.000] Type: GeometryChangeRequest PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1 PlatformRoll: 1.0 PlatformPitch: 1.0 PlatformYaw: 1.0
[000003.000] Type: GeometryChangeRequest PlatformId: PLATFORM.000001448e73b72a PlatformName:
 AirVehicle1
[000004.000] Type: GeometryChangeRequest RadioId: Radio1 RadioName: Radio1 RadioRoll: 0.0
 RadioPitch: 0.0 RadioYaw: 0.0 AutoTrackPlatformName: AirVehicle2
```

```
[000004.000] Type: GeometryChangeRequest RadioId: Radio1 RadioName: Radio1
```

11.7.2.13 SignalToolScriptEvent (D-Series Only)

Signal Tool Script events are generated for Signal Tool related script events.

Name	Description	
Type	Always SignalToolScriptEvent	Required
Mode	Indicates the mode of the event. Record or Replay.	Required
Action	Indicates the action of the event. Start or Stop.	Required
SubType	Indicates the subtype of the event. Port or Noise.	Required
TriggerImmediate	Indicates if the event should triggered immediately.	Required
TriggerRx	If not null, indicates the receive port of the event.	Required
TriggerTx	If not null, indicates the transmit port of the event.	Required
Length	Contains the length, in samples, of the signal. This field is only used when Mode is set to Record.	Required
Delay	Contains the delay of the event.	Required
SignalFile	Contains the filename for the signal. When Mode is set to Record, the results will be saved here. When Mode is set to Replay the signal will be loaded from here.	Required

Example Output:

```
[000001.000] Type: SignalToolScriptEvent Mode: Record Action: Stop SubType: Noise Port: null
 TriggerImmediate: false TriggerRx: null TriggerTx: null Length: 0 Delay: 0 SignalFile: null
```

11.7.2.14 DelayProfileScriptEvent (D-Series Only)

Delay Profile Script events will set the delay profile for a particular channel at the specified time.

Name	Description	
Type	Always DelayProfileScriptEvent	Required
Radio1Name	Indicates the name of the source radio of the link.	Required
Radio2Name	Indicates the name of the destination radio of the link.	Required
TapCount	Indicates the number of subsequent taps within the event	Required
TapNControl	Value provided per tap that indicates the control mode.	Required
TapNGain	Value provided per tap that indicates the gain.	Required
TapNPhase	Value provided per tap that indicates the phase.	Required
TapNDoppler	Value provided per tap that indicates the doppler.	Required
TapNDelay	Value provided per tap that indicates the delay.	Required

Example Output:

```
[000010.000] Type: DelayProfileScriptEvent Radio1Name: 0 Radio2Name: 1 TapCount: 7
Tap0Control: 0.0 Tap0Gain: -10.0 Tap0Phase: 0.0 Tap0Doppler: 0.0 Tap0Delay: 32.2
Tap1Control: 0.0 Tap1Gain: -20.0 Tap1Phase: 0.0 Tap1Doppler: 0.0 Tap1Delay: 4.0
Tap2Control: 0.0 Tap2Gain: -30.0 Tap2Phase: 0.0 Tap2Doppler: 0.0 Tap2Delay: 8.0
Tap3Control: 0.0 Tap3Gain: 0.0 Tap3Phase: 0.0 Tap3Doppler: 0.0 Tap3Delay: 8.0
Tap4Control: 0.0 Tap4Gain: 0.0 Tap4Phase: 0.0 Tap4Doppler: 0.0 Tap4Delay: 0.0
```

11.8 Delay Profile Format (D-Series Only)

The delay profile for each frequency group and link type is specified in the Group Configuration dialog. This defines the channel impulse response (CIR) and is applied to the channels once at the start of the scenario. As platforms move, this CIR is scaled based on the calculated pathloss between nodes. The delay profile is a comma separated value (CSV) format with up to 20 entries per line (1 per tap supported by your resource profile, see Section 5.3), where the lines contain:

Line 1) the distribution to use for Doppler spread (if available). 0 = disabled, 1-4 = use distribution X

Line 2) the gain multiplier from 0 to 32767

Line 3) the phase from 0 to 65535 (2 PI)

Line 4) the Doppler shift from (-200000 to 200000 Hz)

Line 5) the delay in nanoseconds (maximum is resource profile specific, see Section 5.3).

Subsequent lines are ignored.

The gain multiplier can be calculated based on the formula, where gain is the intended gain (i.e., a negative value since a loss is being applied) in dB. For example, a gain of -3 dB gives a binary gain of 23197.

$$binary_{gain} = min(32767, 32767 * 10^{gain/20})$$

The delay of the first tap must be at least four times the time per sample defined in the resource profile. The number of taps available is dependent on the resource profile.

Below is an example file, with Doppler spread disabled, no Doppler shift, and no phase shift (gain and delay only).

```
0, 0, 0, 0
32767,16384,8192,4096
0, 0, 0, 0
0, 0, 0, 0
138,1000,1000,1000
```

11.9 Signal Recording Format (D-Series Only)

The Signal Tool records and saves signals in a CSV format. Each line of the file contains values in i,q format. The number of i,q sample pairs must be a multiple of 256. The maximum number of samples is defined by the resource profile. The i and q values are from 0 to 1.

Example file (note file length must be a multiple of 256, i.e. this example is too short and is only to demonstrate the per line format)

```
1.000000,0.000000
-0.809015,-0.587786
0.309030,0.951048
0.309030,-0.951048
-0.809015,0.587786
1.000000,0.000000
-0.809015,-0.587786
0.309030,0.951048
0.309030,-0.951048
-0.809015,0.587786
```

Below is an example of Matlab code to generate a recording file of a 16QAM signal for replay in RFnest.

```
% This file generates 16QAM samples with sampling rate of 250 Ms/s and symbol rate of 10Ms/s
clear
%Here we set the sampling frequency
fs = 250e6;
% Here we set the symbol rate. Forexample if the modulation order is 2 and symbol rate is 10Ms/s
% then the bit rate is 10 Mb/s
fss=10e6;
%Center frequency after of the QAM signal in RFnest baseband BW of -100 to 100 MHz.
f_c = 50e6;
% N is the total number of samples generated
N = 256000;
% Here we set the sample time and the number of symbols for QAM symbols. Please note that this is
% before upsampling done after rcosine filtering
Ts = 0:1/fs:1/fs*(N-1);

Modulation_order = 16; %Here we set the modulation order to be QAM

rrcos_pulse_shaping = rcosine(fss,fs,'sqrt',.5);
pulse_shape_length = length(rrcos_pulse_shaping);
SER =[];
SER_theory =[];


% x is a random sequence of integers from 0 to Modulation_order-1
x = randint(length(Tss),1,Modulation_order);
% y is QAM of x
y=modulate(modem.qammod(Modulation_order),x);

I=real(y);
Q=imag(y);

%% Upsampling
upsamp = zeros(1, fs/fss);
upsamp(1) = 1;

I = kron(I',upsamp);
Q = kron(Q',upsamp);
%% Root raised cosine pulse shaping

I=conv(rrcos_pulse_shaping,I);
Q=conv(rrcos_pulse_shaping,Q);
I=I(1:N);
Q=Q(1:N);
Signal = I+sqrt(-1)*Q;
Signal = Signal .* exp(sqrt(-1)*2*pi*f_c*Ts);
I = real(Signal);
Q= imag(Signal);
X=[I' Q'];
csvwrite('rrcosine_QAM.csv', X);
```

11.10 Import Scenario Script

This menu allows the user to define platforms and radios. The file format allows assignment of radios to ports to be specified. Most importantly, time-varying positions and/or channel parameters can be specified.

- Use RFview to define the necessary platform types and radio types used in the scenario script file to be imported.
- Select 'Import' then 'Scenario Script' from the 'Scenario' menu.

- Browse to the location of the scenario script file. Scenario script files have an `.sscpt` extension.
- Click 'Import' to load scenario script or 'Cancel' to cancel.

Scenario script files are comma separated value text files consisting of one or more lines. When making a scenario script file, ensure the file's extension is `.sscpt`. Each line contains a timestamp, a command, and the parameters for that command. Valid commands, their parameters, and their meaning are explained below. For more example scenario script files, see the RFview Validation report.

Note that the scenario script file must be sorted by time, i.e. each line's `<time>` parameter must be greater than or equal to the previous line's. Note also that when a scenario script file is imported, any previous update platform and update link commands are automatically purged.

11.10.1 Create Platform

The create platform command is used to instantiate a platform within RFview. The format for this command is:

```
<time>, create platform, <name>, <platform-type>, <lat>, <lon>, <alt>, <roll>, <pitch>, <yaw>
```

- <time> must be 0.
- <name> is arbitrary text which will be displayed as the platform's name.
- <platform-type> is the text name of the already created platform type within the scenario.
- <lat> is the decimal latitude of the platform.
- <lon> is the decimal longitude of the platform.
- <alt> is the decimal altitude of the platform relative to MSL.
- <roll> is the decimal roll of the platform in degrees, -180 to +180.
- <pitch> is the decimal pitch of the platform in degrees, -180 to +180.
- <yaw> is the decimal yaw of the platform in degrees, -180 to +180.

A platform type matching the text entered for any `<platform-type>` parameter must already exist prior to importing the scenario script file. If the platform of the same name already exists, the existing platform will be updated to the type specified.

11.10.2 Create Radio

The create radio command is used to instantiate a radio within RFview. The format for this command is:

```
<time>, create radio, <name>, <radio-type>, <platform-name>, <port-num>
```

- <time> must be 0.
- <name> is arbitrary text which will be displayed as the radio's name.
- <radio-type> is the text name of the already created radio type within the scenario.
- <platform-name> is the name of the already-existing platform to which the radio will be attached.
- <port-num> is the numerical port number where this radio is attached.

A radio type matching the text entered for any `<radio-type>` parameter must already exist prior to importing the scenario script file. Similarly, the platform given in the `<platform-name>` parameter must already exist (but can be created within the same scenario script file). More than one radio may be attached to the same platform. It is recommended that all create radio commands appear after all create radio commands in the file. If a radio of the same name already exists, the existing radio will be updated to the type, platform, and port number specified.

11.10.3 Update Platform

The update platform command is used to move a platform over time. The format for this command is:

```
<time>, update platform, <name>, <lat>, <lon>, <alt>, <roll>, <pitch>, <yaw>
```

- <time> the time in seconds, greater than or equal to zero. A decimal number may be provided, however the precision is dependent on network and processing delays within the software.
- <name> is the name of the already-existing platform to be moved.
- <lat> is the decimal latitude of the platform.
- <lon> is the decimal longitude of the platform.
- <alt> is the decimal altitude of the platform relative to MSL.
- <roll> is the decimal roll of the platform in degrees, -180 to +180.
- <pitch> is the decimal pitch of the platform in degrees, -180 to +180.
- <yaw> is the decimal yaw of the platform in degrees, -180 to +180.

When a platform is moved via an update platform command, the platform will visually move on the map within RFview. A Position Update message will also be generated per the RFnest API. The platform is instantaneously moved to the specified position at the specified time. The parameters for the links between the radios contained on the moved platform and the radios of all other platforms will be recalculated automatically, unless the link is in external mode due to the presence of an update link command for that link.

When the update platform command is used to move a platform during execution of the scenario, the platform will no longer attempt to traverse any predefined route that may have been specified.

Note that it is recommended to specify an update platform command no more than once per second per platform, although for small scenarios this can generally be increased if the software is running on a capable system.

11.10.4 Update Link

The update link command is used to change a link's parameters over time. The format for this command is:

```
<time>, update link, <radio1-name>, < radio2-name>, <tap-count>,
    <main-amplitude>, <main-doppler-shift>, <main-doppler-spread>, <main-delay>,
    <tap1-control>, <tap1-amplitude>, <tap1-phase>, <tap1-doppler>, <tap1-delay>,
    ...,
    <tapn-control>, <tapn-amplitude>, <tapn-phase>, <tapn-doppler>, <tapn-delay>
```

Note that command must appear as a single line in the file.

- <time> the time in seconds, greater than or equal to zero. A decimal number may be provided, however the precision is dependent on network and processing delays within software.

- <radio1-name> and <radio2-name> are the names of the already-existing radios comprising the link.
- <tap-count> is the number of taps in the channel, must be at least 1.
- <main-gain> is the overall (LOS) gain of the channel, decimal from 0 (perfect channel) to -infinity (no channel).
- <main-doppler> is the overall (common) Doppler shift of the channel, in Hz, decimal +/-200,000. May be 0.
- <main-doppler-spread> is a reserved field for future use. 0 should be used.
- <main-delay> is the overall (LOS) delay of the channel. Positive decimal in ns
- <tapX-control> is a reserved field for future use. 0 should be used.
- <tapX-gain> is the gain of tap X, decimal from 0 to -infinity.
- <tapX-phase> is the phase of tap X, decimal from 0 to 2 PI.
- <tapX-doppler> is the Doppler of tap X, in Hz, decimal +/-200,000. May be 0.
- <tapX-delay> is the delay between tap X and the previous tap, Positive decimal in ns.

Note that the number of taps, delay maximums, and Doppler functionality is resource profile specific. Refer to Section 5.3 for the details of your system. If fewer taps are specified than the system is capable of, the gains for any additional taps will be set to zero.

When a link is updated via an update platform command, the updated parameters are sent to the RFnest hardware. Specifically, a Set Delay Profile API message is sent directly to the hardware. A DCU Notification is also sent to the CEC, which in turn sends a Matrix API message to the RFnest hardware to complete the channel update. The link's properties are also updated on the map if the link quality overlay is enabled. The parameters for the currently selected link in the Channel Properties Editor and Channel Impulse Response widget are also updated.

Note that it is recommended to specify an update link command no more than once per second per platform, although for small scenarios this can generally be increased if the software is running on a capable system.

When an update link command is imported for a given link, that link is placed into "external" mode. In this mode, the link can only be modified by the imported update link commands. Editing by the user and calculation by the CEC and any models, including antenna patterns, is disabled. It is assumed the complete modeling of that link has been taken over by the scenario script file. This behavior applies on a per-link basis. To remove a link from external mode, re-import a scenario script file with no update link commands for that link, or re-create the scenario.

The main-* fields apply the specified gain, delay, and Doppler to all taps in addition to the tap's own values. The user may choose to not specify the main-* fields by setting them to 0. Note that if the user prefers to specify a single tap/reflector, it is not sufficient to only specify the main-* fields, at least 1 tap must always be specified.

11.10.5 Set Route

The set route command is used to specify the waypoint traveled by the platform over time. The format for this command is:

```
<time>, set route, <platform-name>, <waypoint-count>,
    <waypoint1-lat>, <waypoint1-lon>, <waypoint1-alt>, <waypoint1-spd>, <waypoint1-delay>,
    ...
    <waypointn-lat>, <waypointn-lon>, <waypointn-alt>, <waypointn-spd>, <waypointn-delay>,
    [<cruise-speed>,]?, <loops>
```

- <time> the time in seconds, greater than or equal to zero. A decimal number may be provided, however the precision is dependent on network and processing delays within the software.
- <name> is the name of the already-existing platform.
- <waypoint-lat> is the decimal latitude of the waypoint.
- <waypoint-lon> is the decimal longitude of the waypoint.
- <waypoint-alt> is the decimal altitude of the waypoint relative to MSL.
- <waypoint-spd> is the speed of the platform in m/s for that waypoint
- <waypoint-delay> is the time to wait, in seconds at that waypoint
- <cruise-speed> indicates the desired cruise speed for the route. This property is only used by KTG mobility.
- <loops> indicates if the route loops back to the first waypoint. This value is 'true' or 'false'.

When a platform route is set via this command, the platform will attempt to travel along the route based on the mobility model specified in the platform type. As the platform moves along the specified route, A Position Update message will also be generated per the RFnest API. The parameters for the links between the radios contained on the moved platform and the radios of all other platforms will be recalculated automatically, unless the link is in external mode due to the presence of an update link command for that link.

The set route command may only be used for platforms with 'Simple' or 'KTG' mobility types. This command may only be used at time 0, and after the initial location of the platform has been specified.

11.11 Import AWSIM Scenario

This menu allows the user to import platforms and routes from an external file that uses the Air Warfare Simulator (AWSIM) output format. The nodes are created automatically, however, the routes can NOT be edited once input into RFview.

1. Select 'Import' then 'AWSIM Scenario' from the 'Scenario' menu.
2. Browse to the location of the AWSIM scenario log file.
3. Click 'Import' to load AWSIM scenario or 'Cancel' to cancel.
    - The imported scenario will instantiate platforms and routes from the AWSIM log, the user will need to assign radios to platform to run.

11.12 Kinematic Trajectory Generator (KTG)

The Kinematic Trajectory Generator (KTG) creates airborne routes based on aircraft attributes, configured platform speed, and configured platform waypoints. KTG allows additional aircraft types to be supplied via Base of Aircraft DAta (BADA) formatted files, an international standard for this type of information. These files must be placed in the config/BADAdata directory. The default BADA files are for generic aircraft not representative of any particular aircraft. BADA files for various existing aircraft are available at https://www.eurocontrol.int/services/bada.

11.13 Command Line Interface

RFview supports a command-line interface for automatically starting scenarios, stopping scenarios, and capturing output, which can be useful for running batch scenarios. For a complete list of supported command-line operations, type `rfview.sh -help\verb`.

```
usage: rfnest
  -help                    Print this message
  -initialize              Initialize the event engine
  -openscenario <file>     Open the scenario
  -resourceprofile <id>    Set the resource profile
  -setplaymode <mode>      Set the play mode, values: [Sandbox, Record,
                            Playback]
  -shutdown                Shutdown the workbench after the scenario
                            completes, or is stopped.
  -start                   Start the scenario
```

12. Built-in test/Self-test and RF Chain Calibration (D-Series Only)

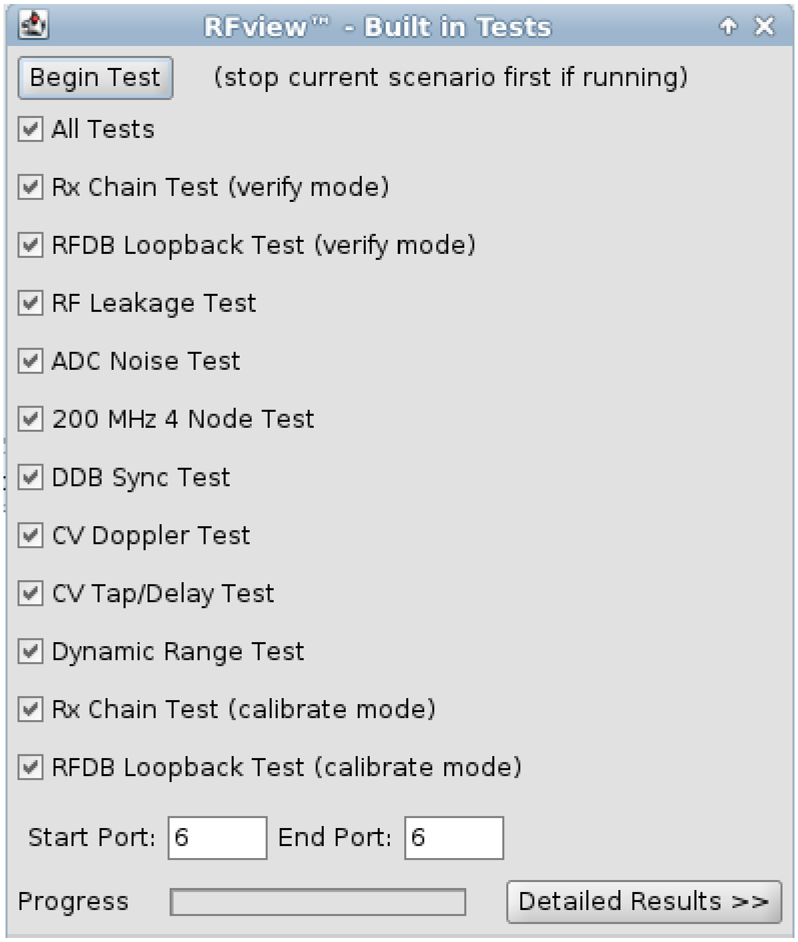
The Built-in Test (BIT) enables the user to check the current functionality of the hardware as well as generate a calibration file. Note that each test requires a specific hardware configuration and RF cabling. In most cases, test should be run individually for this reason.

> *The first four tests are digital only and should be used with resource profile 2 loaded. The remaining tests are RF-based and should be run with resource profile 1 or 3 loaded and the standard hardware_props.xml must be replaced with the contents of the hardware_props.xml.for_bits file and RFview restarted.*

The dialog itself is described below, followed by the meanings and proper setup and execution of each BIT.

12.1 Using the Built-in Test Dialogs

1. Initialize the current scenario. If a scenario is already playing, stop the scenario and then re-initialize the scenario. The BIT menu entry will now be available.
2. Select the built-in self-test utility by clicking on 'Hardware' menu and selecting 'Self-Test'. The following dialog will appear.



3. Check the appropriate checkboxes for the desired tests. The “all tests” checkbox can be used to toggle all tests.

4. Set the range of ports to run the tests on in the ‘Start Port’ and ‘End Port’ fields. The default is to run the test for all ports, which is often the desired action.

5. Click ‘Begin Test’ to run the test(s).

6. The high-level progress and pass/fail status will be displayed for each test selected. Note that the progress bar may not move for several seconds at the start of each test and for several minutes during certain tests. The progress bar may move backward as the number of items to be tested is recalculated periodically.

7. Click on ‘Detailed Results’ to see results.

8. In the event of any failures, the first step is to re-run the test to confirm the failure.

9. The results of the tests are continuously appended to the file logs/test_results.txt. This data can be used for further troubleshooting and is also used as part of the RF chain calibration process. This file can become quite large and can be removed after the test and/or calibration is complete. This file can also be monitored (e.g. using the tail command under Linux) to verify that the test is executing in cases where the progress bar is not moving for an extended period of time.

12.2 Running the DDB Sync Test

This is a **combined digital and RF test** which demonstrates the frequency synchronization among all DDBs in the system. This test takes about 1 minute to execute. While frequency synchronization would only not

occur if the shared clock used by all DDBs failed (unlikely), this test is particularly useful as a quick test of the system in that it requires minimal setup and verifies the ability of the CEB, all DDBs, and the software to communicate correctly.

To run the test, disconnect all radios from the Rx and Tx connectors on the top port on each DDB (i.e. port 0, port 4, etc.). In their place, connect a short RF cable which loops the signal back at each port, in other words port 0's Rx port is now connected to port 0's Tx port, port 4's Rx port is now connected to port 4's Tx port, and so on. No attenuators should be used.

Next, load the test24.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover one or more DDB pairs. Entering start port 0 and end port 23 will test all DDB pairs in a 24-node module and is the normal case.

The test operates by requesting a 200 kHz Doppler shift be applied from the top ports among the DDBs within the selected range. A signal is then automatically digitally injected (similar to the functionality of the Signal Tool) at the top port of one DDB and recorded at the top port of another DDB. At the receiving DDB, the signal loops through the RF chain of the port, exiting the Tx (bottom) connector and entering the Rx (top) connector. The test then verifies that the total shift does not deviate from the expected 200 kHz by more than the resolution of the FFT used by the test (approximately 125 Hz).

12.2.1 Handling Failure

This test will fail if RFview and the CEC are not both running and cannot both communicate to the hardware or the hardware is not running. Note that while the RFview initialization button will not succeed unless the CEB has been successfully communicated with, DDB communication errors may still be present. If certain DDBs fail due to no response, the user should use the Signal Tool to attempt to record a signal from one port on each DDB. If the DDB does not respond at all, communication to/from that DDB is not occuring. **This test will also fail if the user does not attach a loopback RF cable to each top port of the DDBs being tested.** The test will also fail if a scenario besides the test24.xml scenario is loaded. The test may also fail if the CEC's configuration is incorrect, for example if the dopplerincrement parameter is not set correctly, analog mode is enabled, the basic models are disabled and no replacement model is responding to CCR messages, etc. Finally, the test may also fail if the ambient temperature in which the system is operating is too high. It is almost certain that one of those problems is present if the test fails. No other causes are known to be possible.

12.3 Running the CV Doppler Test

This is a digital-only test which verifies the ability of the system to apply a Dopper shift to each channel. This test takes about 15 minutes to execute in a 24-node system. This test verifies that various Doppler shifts (4 kHz, 60 kHz, and 200 kHz) are applied correctly. Note that other Doppler shift values (both positive and negative) can be generated by the system, but only these three values are tested by this test.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. No RF cables need to be connected as this test is digital-only, however a loopback cable can be present on each port and will not affect the test.

Next, load the test24.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the channels being tested. Entering start port 0 and end port 23 will test all 276 channels in a 24-node module and is the normal case. Note that since the channels are symmetric and the FFT used by this test is time-consuming, the channels are only tested in one direction.

The test operates by requesting the three Doppler shifts be applied among all channels between ports within the selected range. The test then verifies that the total shift does not deviate from the expected shift by more than the resolution of the FFT used by the test (approximately 125 Hz).

12.3.1 Handling Failure

This test will fail for the same reasons as the DDB Sync test. It will also fail if any of the ports have an active radio attached to them, which will interfere with the automated process of generating and recording signals through each channel. If a particular channel fails, verify that the receiving port for that channel has no radio attached.

12.4 Running the CV Tap/Delay Test

This is a digital-only test which verifies the ability of the system to apply a delay profile (i.e. a channel impulse response) to each channel. This test takes about 5 minutes to execute in a 24-node system. This test verifies that each tap within each channel is functioning. This test also verifies on a bit by bit basis that each channel exactly matches the expected value and does not contain any digital noise.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. No RF cables need to be connected as this test is digital-only, however a loopback cable can be present on each port and will not affect the test.

Next, load the test24.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the channels being tested. Entering start port 0 and end port 23 will test all 552 channels in a 24-node module and is the normal case. The test verifies the channels in both directions.

12.4.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test. The test may also fail in rare cases if a RF port is not attached to a loopback cable or terminated, i.e. is left open to the air and an ambient RF signal is picked up by that port. In general, since this test is extremely sensitive, if the test fails it is recommended to re-run the test.

12.5 Running the Dynamic Range Test

This is a digital-only test which verifies the ability of the system to apply a gain (i.e. pathloss) to each channel. This test takes about 5 minutes to execute in a 24-node system. This test verifies that a 60 dB digital dynamic range is present in the system. A subset of channels are verified, specifically the channel from each port within the defined range to the next port in the defined range. For the last port, the channel to the previous port is used.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. No RF cables need to be connected as this test is digital-only, however a loopback cable can be present on each port and will not affect the test.

Next, load the test24.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the channels being tested. Entering start port 0 and end port 23 will test 24 channels in a 24-node module and is the

normal case.

12.5.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test.

12.6 Running the RF Leakage Test

This is a RF-only test which verifies the RF signal generated by one Tx port does not excessively leak into a neighboring Rx port. This test takes about 5 minutes to execute. The port range entered by the user is unused, only the leakage into an arbitrary port is considered since each DDB's design is identical.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect a loopback RF cable to each port.

Next, load the test24_full_rate.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range is ignored.

12.6.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test, if RF cables are incorrectly connected, if radios are connected to the system, or a high degree of ambient RF noise is present in the environment.

12.7 Running the ADC Noise Test

This is a RF-only test which verifies the RF signal received on each Rx port does not have excessive noise. This may occur if the port is defective, the system clocks have not synchronized correctly, or the ambient temperature is too high. If the noise at the ADC of a port is too high, then the SNR of the signal may be too low to allow the signal to be correctly received by other radios connected to the system. This test takes about 5 minutes to execute. All ports within the specified range are tested.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect a loopback RF cable to each port or place a 50 Ohm terminator on the Rx connector of each port.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the ports being tested. Entering start port 0 and end port 23 will test 24 channels in a 24-node module and is the normal case.

12.7.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test, if RF cables are incorrectly connected, if radios are connected to the system, or a high degree of ambient RF noise is present in the environment, or the port which failed is defective.

12.8 Manually Testing the System Noise Floor Using a Spectrum Analyzer

This is a RF-only test which verifies the system can generate a very weak output signal. In other words, that the system can cleanly weaken a signal starting from a good SNR to a signal which is well below the receive sensitivity of the radio. This test takes about 5 minutes to execute, is completely manual, and requires a spectrum analyzer with the desired sensitivity.

To run the test, disconnect all radios from the system. Connect the Tx connector of one port to a spectrum analyzer via an RF cable.

Create a scenario where the port being tested is assigned a radio at the desired frequency. The test24.xml scenario is pre-setup for testing any port at 2.4 GHz. Initialize the scenario. Using the Port Status widget, set the output gain of the port to -20 and the noise gain to 0. Using the Signal Tool, replay a very narrow tone (e.g. the provided 3MHz tone file) on the noise path with an immediate trigger condition for the DDB in which the port under test resides.

Locate the signal in the spectrum analyzer. Lower the span and amplitude of the spectrum analyzer as much as possible, while keeping the signal centered. Using the Port Status widget, lower the output gain or noise gain in 10 dB increments, then lower the span/amplitude of the spectrum analyzer. Repeat until the output gain is set to -60 and the noise gain is set to -60, or the desired level. If an attenuator is planned to be added on the output port, this can be added to the cable attached to the spectrum analyzer as well. Ensure that the spectrum analyzer is using the lowest possible sampling bandwidth (usually 1 Hz), allowing the spectrum analyzer to measure the weakest possible signal.

Measure the amplitude of the signal as well as the amplitude of the noise at the edge of the spectrum analyzer's bandwidth being displayed. The difference in amplitudes determines the SNR of the signals output at this level. A clean signal should be visible on the spectrum analyzer down to approximately -140 dBm, or the limit of the spectrum analyzer, whichever is greater.

12.9 Running the 200 MHz 4 node Test

This is a combined RF and digital test which verifies that a 200 MHz signal is supported by the system. A signal is generated at the source port, send through the system, output at the destination port, looped back into the destination port and then measured to verify a 200 MHz signal does not experience more than 6 dB of non-flatness. This test takes about 1 minute to execute. This test must be run using resource profiles 1 or 3. Resource profile 2 is not supported.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect a loopback RF cable onto ports 0 through 3.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. Only ports 0 to 3 are tested.

12.9.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test, if RF cables are incorrectly connected, if radios are connected to the system, or the port which failed is defective.

12.10 Running the Rx Chain Test in Verify Mode

This is a RF-only test which verifies the calibration of the Rx chain on each port. This test takes about 15 minutes to execute for a 24-node module. All ports within the specified range are tested. This test requires an 8-port splitter is available, otherwise the test will take approximately 5 minutes **PER PORT**.

> *NOTE: If the system has not been previously calibrated using the same 8-port splitter and RF cables, the Rx Chain Test must first be run in Calibrate Mode.*

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect the Tx connector on port 0 to the S port on the splitter, and the Rx connectors on ports 0 to 7 to the output ports of the splitter.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the ports being tested. Entering start port 0 and end port 7.

After the test completes, move the output cables on the splitter to the Rx connectors on ports 8 to 15 and re-run with those ports specified as the start and end ports. Repeat the same process for ports 16 to 23.

12.10.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test, if RF cables are incorrectly connected, or if the system is out of calibration. If a few failures occur, manually update the calibration files signals/tests/insertion_gain.csv and signals/tests/insertion_gain_bit.csv to adjust each failed port/frequency in question by the observed error. **Be sure to update both files, as the former is used during system operation and the latter is used to re-verify the calibration using this test.** If the failures do not occur in the frequencies/ports in which the system is being used, they can be ignored. If a significant amount of failures occur, the system should be re-calibrated.

12.11 Running the RFDB Loopback Test in Verify Mode

This is a RF-only test which verifies the calibration of the Tx chain on each port. This test takes about 15 minutes to execute for a 24-node module. All ports within the specified range are tested.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect a loopback cable from the Rx connector to the Tx connector on each port.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the ports being tested. Note that not all gain/frequency combinations are tested, since some combinations create a signal which is too weak for the system to correctly receive from itself.

12.11.1 Handling Failure

This test will fail for the same reasons as the CV Doppler test, if RF cables are incorrectly connected, if the or if the system is out of calibration. If a few failures occur, manually update the calibration files signals/tests/insertion_gain.csv and signals/tests/insertion_gain_bit.csv to adjust each failed port/frequency in

question by the observed error. **Be sure to update both files, as the former is used during system operation and the latter is used to re-verify the calibration using this test.** If the failures do not occur in the frequencies/ports in which the system is being used, they can be ignored. If a significant amount of failures occur, the system should be re-calibrated.

12.12 Running the Rx Chain Test in Calibrate Mode

This is a RF-only test which creates the calibration of the Rx chain on each port. This test takes about one hour to execute for a 24-node module. This test requires an 8-port splitter is available, otherwise the test will take approximately 20 minutes **PER PORT**.

Before or after running the test, use a signal generator and spectrum analyzer to measure the gain of each port of the 8-port splitter and the attached cables for each frequency (120 MHz, 300 MHz, 500 Mhz, . . . , 5900 MHz). These gains should be placed into the signals/tests/rx_chain_offsets.csv file where the first column is the frequency, the second is the gain for port 1 on the splitter, the third for port 2, etc. Ensure that the gain (difference between the signal generator's configured power and the measured port on the spectrum analyzer), and not the output power is entered. The gains should all be negative.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect the Tx connector on port 0 to the S port on the splitter, and the Rx connectors on ports 0 to 7 to the output ports of the splitter.

Delete or rename the file logs/test_results.txt so only the current results will be in this file at the end of the test.

Replace the signals/tests/insertion_gain_bit.csv file with the signals/tests/insertion_gains_blank.csv file.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

The test can now be selected from the Self-Test dialog. The port range entered should cover the ports being tested. Entering start port 0 and end port 7.

After the test completes, move the output cables on the splitter to the Rx connectors on ports 8 to 15 and re-run with those ports specified as the start and end ports. Repeat the same process for ports 16 to 23. The logs/test_results.txt file will now contain the results for all ports.

12.12.1 Handling Failure

This test will always report as failed, as verification of the results can only be done after the output is processed and the test is re-run in Verify Mode. Another BIT should be run prior to this test to verify the system is otherwise operational and RFview can communicate with the CEC and hardware.

12.13 Running the RFDB Loopback Test in Calibrate Mode

This is a RF-only test which creates the calibration of the Tx chain on each port. This test takes about 90 minutes to execute for a 24-node module. All ports within the specified range are tested.

To run the test, disconnect all radios from the Rx and Tx connectors on all ports on each DDB. Connect a loopback cable from the Rx connector to the Tx connector on each port.

Next, load the test24_rf.xml scenario and press the Initialize button in RFview.

Delete or rename the file logs/test_results.txt so only the current results will be in this file at the end of the test. If this file only contains the results from running the Rx Chain Test in Calibrate Mode, this is unnecessary.

Replace the signals/tests/insertion_gain_bit.csv file with the signals/tests/insertion_gains_blank.csv file.

The test can now be selected from the Self-Test dialog. The port range entered should cover the ports being tested. Note that not all gain/frequency combinations are tested, since some combinations create a signal which is too weak for the system to correctly receive from itself.

12.13.1 Handling Failure

This test will always report as failed, as verification of the results can only be done after the output is processed and the test is re-run in Verify Mode. Another BIT should be run prior to this test to verify the system is otherwise operational and RFview can communicate with the CEC and hardware.

12.14 RF Chain Calibration

The system supports the calibration of the input and output RF chains of each port to adjust for variations among ports as well as the frequency response of the RF components. The calibration is stored in two files, signals/tests/insertion_gain.csv for normal system use and signals/tests/insertion_gain_bit.csv for automated verification of the calibration. The calibration process operates by calibrating the loopback gain of each port at each frequency as well as the receive RF chain only of each port using port 0 as a reference. The gain of the transmit RF chain can then be calculated by subtracting the two.

A new calibration file can be created as needed (e.g. because the system is being used in a significantly different temperature than factory calibration is performed, 70F, after a significant amount of time has passed, or the calibration is found to be incorrect).

To recalibrate the system:

1. Run the Rx Chain Test in calibrate mode.
2. Run the RFDB Loopback Test in calibrate mode.
3. The log file can now be processed to create the calibration file.
4. Create the BIT calibration file as follows (assuming the command is run from the `signals/tests/` directory):

   ```
   /path/to/update_insertion_gain.awk insertion_gain_bit.csv ../logs/test_results.txt > \
     insertion_gain_bit.csv.new
   mv insertion_gain_bit.csv.new insertion_gain_bit.csv
   ```

5. Create the calibration file for normal use as follows (again assuming the command is run from `signals/tests/`):

   ```
   /path/to/remove_splitter_from_rx_chain_calib.awk insertion_gain_bit.csv rx_chain_offsets.csv > ↩
       \
     insertion_gain.csv
   ```

   The Rx Chain Test and RFDB Loopback Test should now be re-run in Verify Mode. It is not necessary to re-start RFview.

12.14.1 How the calibration is used

The calibration is used in two places. First, the frequency response of the Tx chain gain is compensated for in the Transmit Power Offset calculation in the Calibration section of the Group Configuration dialog on a per-frequency group basis. Second, the per-node variation in the Tx Chain gain is used to apply a port-specific automatic adjustment to the Output Gain applied in response to the Output Attenuation requested in the Group Configuration dialog. The Rx chain calibration is only used to assist in the calculation of the Tx chain based on the difference between the loopback gain and Rx gain. The Auto-level feature of the Signal Tool automatically adjusts the Input Gain to compensate for the combined per-port variation in the Rx chain and the attached radio's transmit power (many radios transmit consistently at a slightly different power than indicated in their configuration/specifications).

13. Externally Controlling the RF environment

RFnest™ has several ways to control and specify the RF environment. In the typical case, the RFview generates Position Update messages when nodes move (either due to their configured behavior/waypoints or the user moving the nodes on the map). The CEC listens for these messages and generates a CCR request for all channels of each node which has moved. The channel calculation thread within the CEC receives these requests and sends CCR responses with the recalculated channel values. These updated values are then sent from the CEC to the RFnest™ hardware in the form of a channel matrix update. If fading is configured, updates may be sent to the hardware periodically even if nodes are stationary. To temporarily over-ride the values calculated by the CEC, the user can use the channel properties view to manually set the values of one or more channels.

Unused channels by default are set to full attenuation by RFview. There could be unexpected behavior if all channels are set to zero attenuation when using the A-Series, as leakage might occur across the internal RF splitters. For best performance unused channels should be set to a minimum of 15 to 20 dB attenuation in the A-Series.

To control the scenario in a more flexible way, the following approaches can be used:

1. Use RFview's import function to import a time-varying scenario described in a file. This function is flexible and allows creation of detailed scenarios with no user code development or manual packet creation required. RFview defines a "scenario script" file format which allows for initial import of the platforms and radios as well as time-varying definition of platform positions and/or channel parameters. See Section 11.10 for more details. RFview also allows time-varying platform positions to be imported in AWSIM format. See Section 11.11 for more details.
2. Send RFnest Position Updates or EMANE Position Updates via another tool, such as CORE (Common Open Research Emulator) or various other EMANE-compatible tools. The channel values are then updated via the CEC as the nodes move.
3. The CEC's default channel calculation thread can be disabled by setting basicmodel to off in cec.xml and an alternate implementation that listens for CCR requests and responds with CCR the desired responses can be provided by the user. Each time a node moves, the CEC will generate the necessary CCR requests and maintain necessary state for storing node positions and interacting with the hardware, requiring only the implementation of the desired channel model.
4. CCR responses with request_id = 1 can be sent unsolicited to the CEC when a channel value should be changed. As with option #2, basicmodel should be set to off in cec.xml.

5. RFview can be used to manually set the value of one or more channels via the channel properties view. Any channel for which manual control is checked will use the GUI provided value and the CEC will ignore any values calculated received via CCR requests or otherwise calculated by the CEC.

6. DCU notifications (typically generated by the channel properties view) can be sent to the CEC by software other than RFview. DCU notifications provide a higher level description of the channel properties and the CEC handles necessary translation into the format required by the RFnest™ hardware. The manual control should be set to true to prevent the CEC from calculating values for channels controlled via this method.

7. Channel matrix updates can be sent directly to the RFnest™ hardware, completely bypassing the CEC and RFview. This allows complete control of the channel values, but the CEC and RFview will not be aware of these updates and 1) will not display the current channel state correctly, and 2) the CEC should be disabled, for example by setting `basicmodel` to `off` and not running any software that generates CCR responses.

14. Troubleshooting/FAQ

14.1 Select Resource Profile prompt appears or RFview does not communicate with the hardware

When RFview is started, a prompt may appear indicating the hardware was not detected and asking the user to select a resource profile. If RFview is being run without the hardware (e.g. to define a scenario, analyze recorded data from a prior run, demonstrate a recorded run, etc.) the user should select the resource profile that was or will be used to create the scenario. RFview will then operate as if the hardware is configured with the chosen resource profile. If the user is intending to run RFview with the hardware, the presence of this prompt is an indication that RFview did not receive a response when the hardware was queried. Check that all boards have only their green LED lit on the front panel. If this is the case, then the network connectivity between the hardware and the PC, or the multicast route on the PC, is not correct.

If the A-Series hardware is being used and this dialog appears, the system is incorrectly configured. Replace the `hardware_props.xml` with the original file provided in the A-Series RFview distribution.

14.2 CEC/Internal model test

The CEC has a built-in test mode, to be used if the CEC does not appear to be functioning correctly (e.g., not receiving/sending positions updates, model calculations are not being received, etc.) or to determine whether the fading model parameters achieve the desired result. To activate this mode, set `testmode` to `on` and `processpositionevents` to `on` in `cec.xml`. Then invoke the CEC via the `run.sh` script. After a few seconds, the CEC will print out a table of the calculated pathloss and gain values for various distances (20 entries by default). This validates that the CEC is able to send/receive EMANE position events, send/receive CCR messages, and in general is operating correctly. Note that for those modes to work correctly, `eventservicedevice` in `eventdeamon.xml` must be set to `lo` (the loopback device), or the network stack must be configured to receive multicast packets from itself (a platform-dependent setting), since EMANE position events are being sent from the CEC to itself.

14.3 Wireshark Plugin

A RFnest plugin for Wireshark is available for 64-bit Windows and 64-bit Linux as an installer/package or as a standalone dll/so file. This plugin is useful to determine the API messages send/received by various system components, debug third party tools interacting with them, and understand exactly how the system is operating in a given configuration. The plugin works for Wireshark as well as tshark.

You can copy the `.dll` (windows) or `.so` (linux) into the wireshark plugins directory manually. This works if you have a previous version of the plugin or just a plain version of wireshark you want to add the plugin with. Wireshark version 1.10.X or higher is suggested. The `.dll` will work on 32 or 64 bit (depending on the file used) Windows systems. The `.so` will work on Ubuntu and Fedora, and possibly similar Linux operating systems. To determine the version of the plugin you are currently running or if the plugin is loaded, in Wireshark go to Help, About, Plugins.

Once the RFnest plugin is loaded, RFnest messages will be automatically decoded. To only show RFnest messages, enter rfnest in the filter box in Wireshark. To only show messages of a given type, enter `rfnest.type == #` where # is the type code (e.g., 150). One useful filter is `rfnest.type != 128` which shows all RFnest messages **except** for those sent continuously by the hardware.

15. Acronyms

(V)AC (Voltage) Alternating Current.

(V)DC (Voltage) Direct Current.

A2D Analog to Digital.

ATCA Advanced Telecom Computing Architecture.

BIT Built In Tests.

BNC Barrel Nut Connector.

BW Bandwidth.

CCA Circuit Card Assembly.

CD Compact Disc.

CDL Common Data Link.

CEB Channel Emulation Board.

CEC Channel Emulation Controller.

CORE Common Open Research Emulator.

COTS Commercial Off the Shelf.

D2A Digital to Analog.

DDB Digital Daughter Board.

DoD Department of Defense.

EMANE Extendable Mobile Ad-hoc Network Emulator.

ESD Electostatic Discharge.

FPGA Field Programmable Gate Array.

GB GigaByte.

GUI Graphical User Interface.

IF Intermediate Frequency.

IP Internet Protocol.

LED Light Emitting Diode.

LO Local Oscillator.

PC Personal Computer.

RF Radio Frequency.

RUT Radio(s) Under Test.

SMA Sub-Miniature A.

Intelligent Automation, Inc.

15400 Calhoun Drive, Suite 190, Rockville, MD 20855 | Phone: 301-294-5200 | Fax: 301-294-5201